



TUGAS AKHIR - KS 141501

**RANCANG BANGUN PERANGKAT LUNAK UNTUK
PENDETEKSIAN TOPIK KONSELING TEXT
MENGUNAKAN PEMODELAN GAUSSIAN LATENT
DIRICHLET ALLOCATION (STUDI KASUS : RILIV)**

***SOFTWARE DEVELOPMENT FOR DETERMINING
TOPICS OF TEXT COUNSELING USING GAUSSIAN
LATENT DIRICHLET ALLOCATION MODELING (CASE
STUDY : RILIV)***

**ANUGRAH DWIATMAJA PUTRA
NRP 05211440000080**

**Dosen Pembimbing :
Renny Pradina K., S.T., M.T., SCJP**

**DEPARTEMEN SISTEM INFORMASI
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember
Surabaya 2018**



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - KS 141501

RANCANG BANGUN PERANGKAT LUNAK UNTUK PENDETEKSIAN TOPIK KONSELING TEXT MENGUNAKAN PEMODELAN GAUSSIAN LATENT DIRICHLET ALLOCATION (STUDI KASUS : RILIV)

ANUGRAH DWIATMAJA PUTRA
NRP 05211440000080

Dosen Pembimbing :
Renny Pradina K., S.T, M.T., SCJP

DEPARTEMEN SISTEM INFORMASI
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember
Surabaya 2018



ITS

Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - KS 141501

***SOFTWARE DEVELOPMENT FOR DETERMINING
TOPICS OF TEXT COUNSELING USING GAUSSIAN
LATENT DIRICHLET ALLOCATION MODELING (CASE
STUDY : RILIV)***

**ANUGRAH DWIATMAJA PUTRA
NRP 0521144000080**

**SUPERVISOR :
Renny Pradina K., S.T, M.T., SCJP**

**DEPARTEMEN SISTEM INFORMASI
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember**

LEMBAR PENGESAHAN

RANCANG BANGUN PERANGKAT LUNAK UNTUK PENDETEKSIAN TOPIK KONSELING TEXT MENGUNAKAN PEMODELAN GAUSSIAN LATENT DIRICHLET ALLOCATION (STUDI KASUS : RILIV)

TUGAS AKHIR

Disusun Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada

Departemen Sistem Informasi
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember

Oleh:

ANUGRAH DWIATMAJA PUTRA
NRP. 05211440000080

Surabaya, 16 Juli 2018

**KEPALA
DEPARTEMEN SISTEM INFORMASI**

Dr. Ir. Aris Tjahyanto. M.Kom
NIP. 19650310 199102 1 001

LEMBAR PERSETUJUAN

RANCANG BANGUN PERANGKAT LUNAK UNTUK PENDETEKSIAN TOPIK KONSELING TEXT MENGUNAKAN PEMODELAN GAUSSIAN LATENT DIRICHLET ALLOCATION (STUDI KASUS : RILIV)

TUGAS AKHIR


**Disusun Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada**

**Departemen Sistem Informasi
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember
Oleh :**


**ANUGRAH DWIATMAJA PUTRA
NRP. 05211440000080**

**Disetujui Tim Penguji : Tanggal Ujian : 9 Juli 2018
Periode Wisuda : September 2018**

Renny Pradina K., S.T, M.T., SCJP


(Pembimbing I)

Nur Aini R., S.Kom., M.Sc.Eng., Ph.D


(Penguji I)

Radityo Prasetyanto W., S.Kom., M.Kom.


(Penguji II)

RANCANG BANGUN PERANGKAT LUNAK UNTUK PENDETEKSIAN TOPIK KONSELING TEXT MENGUNAKAN PEMODELAN GAUSSIAN LATENT DIRICHLET ALLOCATION (STUDI KASUS : RILIV)

Nama Mahasiswa : Anugrah Dwiatmaja Putra
NRP : 0521 14 4000 0080
Jurusan : Sistem Informasi FTIK-ITS
Pembimbing 1 : Renny Pradina K., S.T, M.T., SCJP

ABSTRAK

Riliv merupakan platform konseling online yang menghubungkan pengguna dengan masalah pribadi kepada psikolog secara online. Sebagai perusahaan penyedia jasa dan layanan konseling, Riliv berusaha untuk mengutamakan kualitas pelayanan dan kepuasan pelanggan. Riliv memiliki permasalahan dimana para psikolog menghabiskan banyak waktu di awal konseling untuk memahami topik permasalahan dari pengguna sehingga menyebabkan pengguna harus menunggu respon awal dari psikolog dengan lebih lama. Maka dari itu, penelitian ini akan melakukan analisis topic modeling pada konseling yang telah diselesaikan oleh Riliv untuk mengetahui topik-topik apa saja yang sering disampaikan oleh penggunanya.

Analisis topik dilakukan menggunakan metode Gaussian Latent Dirichlet Allocation dan dievaluasi dengan pengujian perplexity untuk mengetahui kualitas dari model dalam melakukan generalisasi terhadap dokumen. Pengukuran uji koherensi dilakukan dengan perhitungan Pointwise Mutual Information (PMI) untuk menganalisis tingkat kesamaan semantic antara kata-kata yang terdapat didalam topik. Pemodelan topik juga dilakukan dengan menggunakan metode Latent Dirichlet Allocation (LDA) sebagai pembanding dari hasil Gaussian LDA. Kedua model selanjutnya dibandingkan dengan mengukur tingkat coherence masing-masing model menggunakan perhitungan Pointwise Mutual Information (PMI). Probabilitas kata-kata dalam setiap topik yang

dihasilkan dari pemodelan terbaik dengan metode Gasusian LDA dan LDA selanjutnya digunakan untuk melakukan pendeteksian terhadap probabilitas topik-topik pada konseling yang baru.

Berdasarkan hasil eksperimen pemodelan topik yang dilakukan, dapat disimpulkan bahwa skenario data dengan stemming pada metode pemodelan Gaussian LDA 25 topik menghasilkan model dengan nilai coherence tertinggi dengan perhitungan PMI, yaitu sebesar 3.0286.

Kata Kunci: Topic Modeling, Gaussian Latent Dirichlet Allocation, Konseling, Riliv, Perplexity, Topic Coherence

SOFTWARE DEVELOPMENT FOR DETERMINING TOPICS OF TEXT COUNSELING USING GAUSSIAN LATENT DIRICHLET ALLOCATION MODELING (CASE STUDY : RILIV)

Student Name : Anugrah Dwiatmaja Putra
NRP : 05211440000080
Department : Sistem Informasi FTIK-ITS
Supervisor 1 : Renny Pradina K., S.T, M.T., SCJP

ABSTRACT

Riliv is online counseling platform that connecting people with personal problems to psychologist by online. As a provider of counseling services, Riliv trying to give priority of services quality and customer satisfaction. Riliv has a problem that psychologists a lot of time in the early counseling to understand the topic of the problem from user thus causing the user to wait for an earlier response from the psychologist for longer. Therefore, this research will do the topic modeling analysis on counseling that has been completed by Riliv to find out topics are often delivered by users.

The topic analysis done using the Gaussian Latent Dirichlet Allocation method and evaluated by the perplexity test to determine the quality of the model in generalizing the document. Measurement of coherence test is done by calculation Pointwise Mutual Information (PMI) for the semantic degree of similarity between the words contained in the topic. Topic modeling is also done using Latent Dirichlet Allocation method as a comparison of Gaussian LDA result. The two models are compared by measuring the coherence level of each model using PMI. The probability of words in each of the topics resulting from Gaussian LDA and LDA methods will be used to detect the probability of topics in new counseling. Based on the experimental result of topic modeling, concluded that the data scenario with stemming one the Gaussian LDA 25

topics modeling method resulted have highest coherence value by PMI, which is 3.0286.

Keywords: Topic Modeling, Gaussian Latent Dirichlet Allocation, Counseling, Riliv, Perplexity, Topic Coherence

KATA PENGANTAR

Puji dan syukur penulis tuturkan ke hadirat Allah SWT, Tuhan Semesta Alam yang telah memberikan kekuatan dan hidayah-Nya. Shalawat serta salam tidak lupa penulis curahkan kepada Sayyidina Muhammad SAW beserta keluarga, para sahabat dan umatnya.

Terima kasih penulis sampaikan kepada pihak-pihak yang telah mendukung, memberikan saran, motivasi, semangat, dan bantuan baik berupa materiil maupun moril demi tercapainya tujuan pembuatan tugas akhir yang merupakan salah satu syarat kelulusan pada Departemen Sistem Informasi, Fakultas Teknologi Informasi dan Komunikasi, Institut Teknologi Sepuluh Nopember Surabaya. Tugas akhir ini tidak akan pernah terwujud tanpa bantuan dan dukungan dari berbagai pihak yang sudah meluangkan waktu, tenaga, dan pikirannya. Secara khusus penulis akan menyampaikan ucapan terima kasih yang sebanyak-banyaknya kepada:

1. Bapak Sukarlan dan Ibu Sri Utami selaku kedua orang tua serta Arika Prihastanti Sutami selaku kakak kandung dan Alpha Sutha Media selaku adik kandung dari penulis yang tiada henti memberikan dukungan dan semangat secara lahir dan batin.
2. Keluarga besar penulis di Desa Kemuning Sambit Ponorogo yang tiada henti memberi dukungan dan mendoakan penulis selama masa perkuliahan.
3. Ibu Renny Pradina Kusumawardani., S.T., M.T., SCJP, selaku dosen pembimbing dan sebagai narasumber yang senantiasa meluangkan waktu dan tenaga, memberikan ilmu dan petunjuk dengan sabar yang sangat berarti bagi penulis, serta terus memberikan motivasi dan nasehat untuk kelancaran pengerjaan tugas akhir.
4. Bapak Dr.Eng. Febriliyan Samopa, S.Kom., M.Kom. selaku dosen wali yang selalu membantu penulis dalam masa perkuliahan sejak mahasiswa baru hingga tugas akhir.

5. Ibu Nur Aini Rakhmawati, S.Kom., M.Sc.Eng., Ph.D., Ibu Irmasari Hafidz, S.Kom., M.Sc., serta Bapak Radityo Prasetyanto W., S.Kom., M.Kom. selaku dosen penguji yang telah memberikan saran dan kritik untuk perbaikan tugas akhir.
6. Seluruh dosen Departemen Sistem Informasi ITS yang telah memberikan ilmu yang bermanfaat kepada penulis.
7. Beasiswa Karya Salemba Empat yang telah memberikan kesempatan kepada penulis dalam mengembangkan kemampuan manajerial dan bantuan materiil selama satu tahun perkuliahan di Institut Teknologi Sepuluh Nopember.
8. Briliyand Yusuf Al Hanif, Rizqi Mamluatul, dan Hasna Rosida Putri selaku teman SMA yang memberi dukungan dan semangat yang sangat berarti bagi penulis.
9. Ahmad Choirun Najib selaku sahabat terdekat di kampus yang telah memberikan dukungan yang sangat berarti bagi penulis.
10. Baiq Zuyyina Hilyatur Rozaliya, Prasetyo Wahyu Adi W, Adrian Afnandika, Dewangga Prasetya Praja, Alden Delfian Wattimena, Ahmad Arif Samudro, dan Hans Juno Panjaitan selaku mahasiswa bimbingan tugas akhir dari Bu Renny yang berjuang bersama penulis pengerjaan tugas akhir.
11. Siti Oryza Khairunnisa, Stezar Priansya, dan Nikolaus Herjuno S.D.A yang memberikan pencerahan secara praktek dan teori terkait pengerjaan tugas akhir.
12. Audrey Maximilian Herli dan rekan-rekan Riliv yang telah membantu penulis terutama dalam memperoleh data terkait tugas akhir.
13. Dhimas Dwijo Rahminto, Akmal Faza, Muhammad Fikry Hazmi, Guntur Kondang Prakoso, Tubagus Hendro Pramono, Rysma Aditya Widjaja, Azmi Adi Firmansyah, Aprilia Rizki Rakhmawati, Jwalita Galuh Garini, Aldifiati Arfiani, dan Mutiara Ratna Shafura Sudrajad yang senantiasa menghibur dan mendukung penulis dalam pengerjaan tugas akhir.

14. Rekan-rekan HMSI ITS Muda Berkarya, BEM FTIf ITS Ledakan Inovasi, BEM FTIf ITS Presisi Bermanfaat, dan DPM FTIK ITS 2017/2018 yang telah memberikan banyak kenangan dan pembelajaran sangat berharga semasa kuliah.
15. Rekan-rekan Pemandu Infinity dan Phoenix yang telah bersama bekerja dengan penulis dalam menjalankan seluruh kegiatan LKMM di Fakultas Teknologi Informasi dan Komunikasi selama penulis menjadi Pemandu Aktif.
16. Rekan-rekan kos Gebang Wetan No.21 yang mendukung, menemani, dan berjuang bersama penulis dalam pengerjaan akhir.
17. Rekan-rekan mahasiswa ITS Ponorogo 2014 dan IKASMAZA Surabaya yang mendukung dan semangat bagi penulis.

Penyusunan laporan ini masih jauh dari kata sempurna sehingga penulis menerima kritik maupun saran yang membangun untuk perbaikan di masa yang akan datang. Semoga buku tugas akhir ini dapat memberikan manfaat bagi pembaca

Surabaya, 26 Juni 2018

Anugrah Dwiatmaja Putra

Halaman Sengaja Dikosongkan

DAFTAR ISI

ABSTRAK	vi
ABSTRACT	viii
KATA PENGANTAR	x
DAFTAR ISI	xiv
DAFTAR GAMBAR	xviii
DAFTAR KODE	xx
DAFTAR TABEL	xxii
DAFTAR DIAGRAM	xxiv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	3
1.4 Tujuan Penelitian	3
1.5 Manfaat Penelitian	3
1.6 Relevansi	4
BAB II TINJAUAN PUSTAKA	5
2.1 Studi Sebelumnya	5
2.2 Dasar Teori	7
2.2.1 <i>Topic Modeling</i>	7
2.2.2 <i>Latent Dirichlet Allocation (LDA)</i>	8
2.2.3 <i>TF-IDF</i>	10
2.2.4 <i>Gaussian Latent Dirichlet Allocation (Gaussian LDA)</i>	10
2.2.5 Validasi Topik dengan <i>Perplexity</i> dan <i>Topic Coherence</i>	11
2.2.6 <i>Pointwise Mutual Information (PMI)</i>	12
2.2.7 Rilis	13
2.2.8 Laravel Framework	14
BAB III METODOLOGI	17
3.1 Tahapan Pelaksanaan Tugas Akhir	17
3.1.1 Identifikasi Masalah	17

3.1.2 Studi Literatur	18
3.1.3 Mempersiapkan Data	18
3.1.4 Pra-proses Data	18
3.1.5 Pemodelan Topik	26
3.1.6 Validasi Model Topik	27
3.1.7 Implementasi Deteksi Topik Menggunakan Hasil Pemodelan Topik	27
BAB IV PERANCANGAN	29
4.1 Pengambilan Data	29
4.2 Metodologi Implementasi Penilaian.....	29
4.2.1 Mempersiapkan Data	29
4.2.2 Pra-Proses Data	30
4.2.3 Pemodelan Topik dengan Metode <i>Gaussian LDA</i> 32	
4.2.4 Pemodelan Topik dengan Metode <i>Latent Dirichlet Allocation</i>	33
4.2.5 Validasi Model Topik	35
4.2.6 Konstruksi Perangkat Lunak	35
BAB V IMPLEMENTASI	41
5.1 Perangkat Penelitian.....	41
5.2 Mempersiapkan Data.....	42
5.3 Pra-Proses Data	43
5.3.1 <i>Case Folding, Tokenizing, dan Formalize</i>	43
5.3.2 Stemming	45
5.3.3 Stopword Removal.....	46
5.4 Pemodelan Topik dengan Metode <i>Gaussian LDA</i>	47
5.4.1 Pemrosesan <i>Corpus</i>	48
5.4.2 Pemrosesan Vektor Kata	50
5.4.3 Pembentukan Model Topik.....	52
5.4.4 Eksperimen Pemodelan Topik	56
5.5 Pemodelan Topik dengan Metode <i>Latent Dirichlet Allocation</i>	61
5.5.1 Pembentukan <i>Dictionary</i>	62
5.5.2 Pembentukan Dokumen Matriks atau <i>Corpus</i> ...	63
5.5.3 Pembentukan Model Topik.....	63

5.5.4 Eksperimen Pemodelan Topik dengan <i>LDA Model</i>	63
5.6 Validasi Model Topik <i>Gaussian LDA</i>	66
5.7 Validasi Model Topik <i>Latent Dirichlet Allocation</i>	67
5.8 Konstruksi Perangkat Lunak	69
BAB VI HASIL DAN PEMBAHASAN	72
6.1 Mempersiapkan Data	72
6.2 Pra-Proses Data	73
6.3 Pemodelan Topik dengan <i>Gaussian LDA</i>	73
6.3.1 Pembentukan <i>Corpus</i>	74
6.3.2 Pembentukan Vektor Kata	74
6.3.3 Penentuan Jumlah Topik.....	75
6.3.4 Penentuan Jumlah Iterasi	76
6.4 Pemodelan Topik dengan Metode <i>Latent Dirichlet Allocation</i>	77
6.4.1 Pembentukan <i>Dictionary</i>	77
6.4.2 Penentuan Jumlah <i>Passes</i>	78
6.4.3 Penentuan Jumlah Topik.....	79
6.5 Validasi Model Topik <i>Latent Dirichlet Allocation</i>	82
6.6 Validasi Model Topik <i>Gaussian LDA</i>	85
6.7 Pengujian Model dengan <i>Pointwise Mutual Information</i>	87
6.7.1 Analisis Kuantitatif.....	87
6.7.2 Analisis Kualitatif	88
6.8 Pengujian Model pada Aplikasi	89
BAB VII KESIMPULAN DAN SARAN	100
7.1 Kesimpulan	100
7.2 Saran dan Penelitian Selanjutnya	100
DAFTAR PUSTAKA	102
BIODATA PENULIS	104
LAMPIRAN A	A-1
LAMPIRAN B	B-10
LAMPIRAN C	C-23

Halaman Sengaja Dikosongkan

DAFTAR GAMBAR

Gambar 2. 1. Konsep <i>Topic Modeling</i> menurut Blei.	8
Gambar 2. 2. Visualisasi LDA menurut Blei	9
Gambar 2. 3. Ilustrasi <i>Pointwise Mutual Information</i>	13
Gambar 3. 1. Tahapan Pelaksanaan Tugas Akhir	17
Gambar 3. 2 Sub-Aktivitas dari Tahap Pra-Pemrosesan Corpus	19
Gambar 4. 1. Alur Mepersiapkan Data	30
Gambar 4. 2. Alur Tahap Pra-Proses Data	31
Gambar 4. 3. Alur Pembentukan <i>Dictionary</i> dan <i>Corpus</i>	34
Gambar 4. 4. Alur Pembentukan Model Lda	35
Gambar 4. 5.....	33
Gambar 4. 6.....	33
Gambar 4. 7. Perancangan Basis Data Aplikasi Portal Reliever	36
Gambar 4. 8. Alur Penyimpanan <i>Posting</i> dan Penghitungan Probabilitas Topik	39
Gambar 4. 9. Alur Penampilan Data Konseling dan Probabilitas Topik	40
Gambar 5. 1 Antar Muka Aplikasi Halaman Masuk.....	70
Gambar 5. 2. Antar Muka Aplikasi Halaman Daftar Konseling	70
Gambar 5. 3. Antar Muka Aplikasi Halaman Detail Konseling	71
Gambar 5. 4. Antar Muka Aplikasi Halaman <i>Simulator Posting</i>	71

Halaman Sengaja Dikosongkan

DAFTAR KODE

Kode 5. 1. Kode Program Memuat Data dan Menghapus Data yang Sama	43
Kode 5. 2. Kode Program Menginisiasi Kamus <i>Formalize</i> ...	44
Kode 5. 3. Kode Program <i>Case Folding</i> , <i>Tokenizing</i> , Dan <i>Formalize</i>	45
Kode 5. 4. Kode Program Stemming dengan <i>Library Sastrawi</i>	46
Kode 5. 5. Kode Program <i>Stopword Removal</i>	47
Kode 5. 6. Kode Program untuk Menghubungkan Kata Sambung.....	47
Kode 5. 7. Kode Program untuk Memproses Data dan Membentuk <i>Dictionary</i>	62
Kode 5. 8. Kode Program untuk Mengatur Batasan <i>Unique Tokens</i> pada <i>Dictionary</i>	62
Kode 5. 9. Kode Program untuk Pembentukan <i>Corpus</i>	63
Kode 5. 10. Kode Program Pembentukan Model LDA	63
Kode 5. 11. Kode Program Penentuan Jumlah <i>Passes</i>	65
Kode 5. 12. Kode Program Penentuan Jumlah Topik	65
Kode 5. 13. Kode Program Penentuan Jumlah Topik	66
Kode 5. 14. Pembuatan <i>Corpus</i> dan Penghapusan <i>Unique Tokens</i>	48
Kode 5. 15. Inisiasi <i>Variable</i>	49
Kode 5. 16. Proses Menyimpan <i>Vocabulary</i>	50
Kode 5. 17. Proses Konversi Kata Menjadi Vektor.	51
Kode 5. 18. Inisiasi <i>Variable</i>	52
Kode 5. 19. Memperbarui Jumlah Kata dalam Topik	53
Kode 5. 20. Kode Perhitungan Ulang Operasi Pengurangan	53
Kode 5. 21. Kode Perhitungan Ulang Operasi Penambahan..	54
Kode 5. 22. Perhitungan <i>Log of The Probability Density</i>	55
Kode 5. 23. Kode <i>Gibbs Sampling</i>	56
Kode 5. 24. Kode Mengambil <i>log_posterior</i> Tertinggi, Pembaruan Kata, dan Perhitungan Ulang Parameter <i>Gaussian LDA</i>	56
Kode 5. 25. Kode Perhitungan Probabilitas Kata terhadap Dokumen.....	58
Kode 5. 26. Kode Perhitungan Nilai <i>Perplexity</i>	59

Kode 5. 27. Kode Program Menyimpan Variable Pemodelan <i>Gaussian LDA</i>	61
Kode 5. 28. Kode Program <i>Load</i> Model dan Pengambilan Kata Dalam Topik.....	68
Kode 5. 29. Kode Program Penentuan Jumlah Topik	69
Kode 5. 30. Perhitungan Pointwise Mutual Information	67

DAFTAR TABEL

Tabel 2. 1. Studi Sebelumnya.....	5
Tabel 3. 1. Contoh Data Sebelum dan Sesudah Melalui Tahap <i>Case Folding</i>	19
Tabel 3. 4. Contoh Data Sebelum dan Sesudah Melalui Tahap <i>Tokenization</i>	21
Tabel 3. 2. Contoh Data Sebelum dan Sesudah Melalui Tahap <i>Formalize</i>	22
Tabel 3. 3. Contoh Data Sebelum dan Sesudah Melalui Tahap <i>Stemming</i>	24
Tabel 3. 5. Contoh Data <i>Stopwords</i>	25
Tabel 3. 6. Contoh Data Sebelum dan Sesudah Melalui Tahap <i>Stopwords</i>	25
Tabel 5. 1. Spesifikasi Komputer	41
Tabel 5. 2. Teknologi yang Digunakan Untuk Mengembangkan Model	41

Halaman Sengaja Dikosongkan

DAFTAR DIAGRAM

Diagram 6. 1. Nilai <i>Perplexity</i> Jumlah Topik Uji <i>Stemming</i> ..	75
Diagram 6. 2. Nilai <i>Perplexity</i> Jumlah Topik Uji Tanpa <i>Stemming</i>	76
Diagram 6. 3. Nilai <i>Perplexity</i> 25 Topik <i>Stemming</i> dengan 30 iterasi.	77
Diagram 6. 4. Analisis Nilai <i>Perplexity</i> untuk Penentuan Jumlah <i>Passes</i>	78
Diagram 6. 5. Rata-rata nilai <i>perplexity</i> dalam 30 kali percobaan	79
Diagram 6. 6. Standar Deviasi nilai <i>perplexity</i> dalam 30 kali percobaan	80
Diagram 6. 7. Rata-rata nilai <i>perplexity</i> dalam 30 kali percobaan	81
Diagram 6. 8. Standar Deviasi nilai <i>perplexity</i> dalam 30 kali percobaan	82
Diagram 6. 9. Rata-rata nilai <i>topic coherence</i>	83
Diagram 6. 10. Standar deviasi nilai <i>topic coherence</i>	83
Diagram 6. 11. Nilai <i>topic coherence</i> tertinggi	84
Diagram 6. 12. Nilai <i>topic coherence</i> terendah.....	84
Diagram 6. 13. Rata-Rata Nilai <i>Topic Coherence</i> 25 Topik dengan <i>Pointwise Mutual Information</i>	86
Diagram 6. 14. Standar Deviasi Nilai <i>Topic Coherence</i> 25 Topik dengan <i>Pointwise Mutual Information</i>	86
Diagram 6. 15. Nilai Rata-Rata PMI Model pada Masing-Masing Skenario.....	88

Halaman Sengaja Dikosongkan

BAB I

PENDAHULUAN

Pada bab pendahuluan akan diuraikan proses identifikasi masalah penelitian yang meliputi latar belakang masalah, perumusan masalah, batasan masalah, tujuan tugas akhir, manfaat kegiatan tugas akhir, dan relevansi terhadap pengerjaan tugas akhir. Berdasarkan uraian pada bab ini, harapannya gambaran umum permasalahan dan pemecahan masalah pada tugas akhir dapat dipahami.

1.1 Latar Belakang

Riliv merupakan platform konseling online yang menghubungkan orang-orang dengan masalah pribadi kepada psikolog, melalui online chat, dan panggilan suara atau video call. Misi Riliv adalah membuat terapi yang mudah dan terjangkau untuk meningkatkan kesejahteraan masyarakat. Semua psikolog Riliv merupakan psikolog berpengalaman dan mendapat izin dari Himpunan Psikolog Indonesia (HIMPSI) dan memiliki latar belakang yang luas dalam menangani depresi terhadap masalah perkawinan. Untuk memperluas layanan yang dimiliki, Riliv terus memperkenalkan layanannya ke masyarakat luas. Saat ini, Riliv telah menyelesaikan lebih dari 150.000 konseling yang ada di Indonesia dengan dibantu lebih dari 300 psikolog profesional [1].

Sebagai perusahaan penyedia jasa dan layanan konseling, Riliv berusaha untuk mengutamakan kualitas pelayanan dan kepuasan pelanggan. Setiap pesan dari pengguna merupakan hal yang penting, sehingga para psikolog dari Riliv dituntut untuk memberikan respon secara cepat dan tepat. Adapun pesan yang dikirimkan merupakan permasalahan pribadi yang berbeda setiap pengguna, sehingga perlu penanganan khusus dari setiap permasalahan oleh psikolog yang memiliki kemampuan di bidang tersebut.

Saat ini, Riliv menggunakan aplikasi dan LINE Messenger bagi konseling berbayar dan aplikasi Riliv bagi konseling gratis. Untuk memulai konseling, user diminta untuk

menjawab beberapa pertanyaan yang diajukan tentang data pribadi serta kategori masalah yang ingin disampaikan, kemudian pengguna menjelaskan permasalahan yang dihadapinya dengan pesan. Pesan yang telah dikirimkan selanjutnya dipahami terlebih dahulu oleh psikolog untuk mengetahui dan melakukan validasi topik permasalahan yang dihadapi oleh pengguna. Namun, pemahaman yang dilakukan oleh psikolog memerlukan waktu yang cukup banyak. Adapun kategori konseling yang disediakan oleh Riliv dirasa masih belum merepresentasikan inti permasalahan yang dihadapi pengguna.

Oleh karena itulah, penelitian ini menawarkan solusi dalam melakukan analisis *topic modeling* terhadap permasalahan yang dihadapi pengguna pada konseling. Analisis *topic modeling* digunakan untuk mengetahui topik-topik yang sering muncul beserta distribusi kata dari topik tersebut, sehingga memudahkan Aplikasi Psikolog dalam mendeteksi topik dari setiap permasalahan yang dihadapi pengguna dalam konseling. Penelitian ini menggunakan metode *Gaussian Latent Dirichlet Allocation* atau yang selanjutnya disebut dengan Gaussian LDA. Metode Gaussian LDA merupakan salah satu bentuk metode text mining untuk menemukan pola tertentu dari sebuah dokumen berdasarkan beberapa macam topik yang berbeda.

Hasil dari analisis *topic modeling* menggunakan Gaussian LDA akan digunakan untuk mendeteksi topik dari penjelasan permasalahan oleh pengguna serta menampilkan sebuah visualisasi berupa dashboard pada website yang dapat mengetahui trend dari topik-topik yang sering dibahas untuk membantuk pengambilan keputusan.

1.2 Rumusan Masalah

Berdasarkan uraian latar belakang, maka rumusan permasalahan yang menjadi fokus dan akan diselesaikan dalam tugas akhir ini antara lain :

1. Bagaimana melakukan *topic modeling* untuk menganalisis topik-topik apa yang sering muncul pada layanan konseling text dari Riliv?
2. Bagaimana mengukur tingkat koherensi topik yang menjadi luaran *topic modeling*?
3. Bagaimana model hasil topic modeling dalam mendeteksi topik terhadap data baru?

1.3 Batasan Masalah

Berikut batasan masalah dalam pengerjaan tugas akhir ini berdasarkan pada penguraian rumusan masalah di atas :

1. Studi kasus yang digunakan berfokus pada Riliv.
2. Data yang digunakan merupakan data *posting* konseling dari layanan konseling text Riliv dengan pengguna dari bulan Januari 2017 sampai Desember 2017 dengan bahasa Indonesia.

1.4 Tujuan Penelitian

Berikut tujuan dari tugas akhir ini berdasarkan pada pemaparan latar belakang dan rumusan masalah di atas :

1. Melakukan analisis *topic modeling* untuk mengetahui topik-topik permasalahan yang sering dibahas pada konseling di Riliv.
2. Melakukan validasi pada luaran dari topic modeling untuk mengukur kinerja metode Gaussian LDA dan LDA pada dokumen.
3. Membuat rancang bangun Implementasi Deteksi Topik (Portal Reliever) untuk memudahkan psikolog melakukan sesi konseling dengan pengguna.

1.5 Manfaat Penelitian

Manfaat yang diharapkan dapat diperoleh dari tugas akhir ini adalah:

Bagi Penulis

1. Memahami metode *topic modeling* dengan algoritma *Gaussian Latent Dirichlet Allocation (Gaussian LDA)* dan *Latent Dirichlet Allocation (LDA)* serta mampu menerapkannya pada studi kasus layanan konseling Riliv.
2. Melakukan eksperimen untuk mengetahui kinerja pemodelan *Gaussian Latent Dirichlet Allocation (Gaussian LDA)* dan *Latent Dirichlet Allocation (LDA)* dengan karakteristik data konseling Riliv.

1.6 Relevansi

Tugas akhir ini berhubungan dengan penerapan mata kuliah Algoritma dan Pemrograman, Matematika Diskrit, Sistem Cerdas, dan Penggalan Data dan Analitika Bisnis yang merupakan mata kuliah bidang keilmuan Laboratorium Akuisisi Data dan Diseminasi Informasi.

BAB II TINJAUAN PUSTAKA

Pada bab ini akan membahas penelitian sebelumnya yang berhubungan dengan tugas akhir dan teori-teori yang berkaitan dengan permasalahan tugas akhir.

2.1 Studi Sebelumnya

Tabel 2.1 menampilkan daftar penelitian sebelumnya yang mendasari tugas akhir ini.

Tabel 2. 1. Studi Sebelumnya

1. Gaussian LDA for Topic Models with Word Embedding
Penulis/Tahun/Sumber : Rajarshi Das, Manzil Zaheer dan Cris Dyer
Metode : - <i>Gaussian Latent Dirichlet Allocation (Gaussian LDA)</i>
Kesimpulan : Pada penelitian ini dilakukan analisis topik dengan teknik analisis <i>topic modeling</i> memanfaatkan <i>word embedding</i> sebagai representasi vektor setiap kata pada saat training dilakukan. Jumlah topik 15 dipilih berdasarkan hasil penilaian <i>Pointwise Mutual Information (PMI)</i> untuk mengukur nilai koherensi [3].
2. Latent Dirichlet Allocation (LDA)
Penulis/Tahun/Sumber :

David M. Blei, Andrew Y. Ng dan Michael I. Jordan
Metode : <i>- Latent Dirichlet Allocation</i>
Kesimpulan : <p>Pada penelitian ini dilakukan analisis topik pada data berupa teks. Penelitian ini menyajikan sebuah framework baru untuk melakukan pemodelan struktur dari pada topik dalam dokumen dan data diskrit lainnya [2].</p>
3. Novel Use of Natural Language (NLP) to Predict Suicidal Ideation and Psychiatric Symptoms in Text-Based Mental Health Intervention in Madrid
Penulis/Tahun/Sumber : <p>Benjamin L. Cook, Ana M. Progovac, Pei Chen, Brian Mullin, Sherry Hou, Enriq Baca-Garcia.</p>
Metode : <p>- Menggunakan metode komputasi seperti NLP yang dikombinasikan dengan teknik prediksi dalam <i>machine learning</i> dengan memanfaatkan data dari <i>Electronic Health Records</i>.</p>
Kesimpulan : <p>Penelitian ini bertujuan untuk melakukan analisis pada pesan pendek dengan menggunakan NLP atau analitikal mechine learning. Terdapat 1,453 participant dalam penelitian ini untuk dijadikan objek observasi. 73% dari participants memiliki rata-rata Skor GHQ-12 ≥ 4.</p>

4. Large-scale Analysis of Counseling Conversation: An Application of Natural Language Processing to Mental Health

Penulis/Tahun/Sumber :

Tim Althoff, Kevin Clark, Jure Leskovec

Metode :

- *Sequence-based conversation models, message clustering dan psycholinguistic-inspired word frequency analyses.*

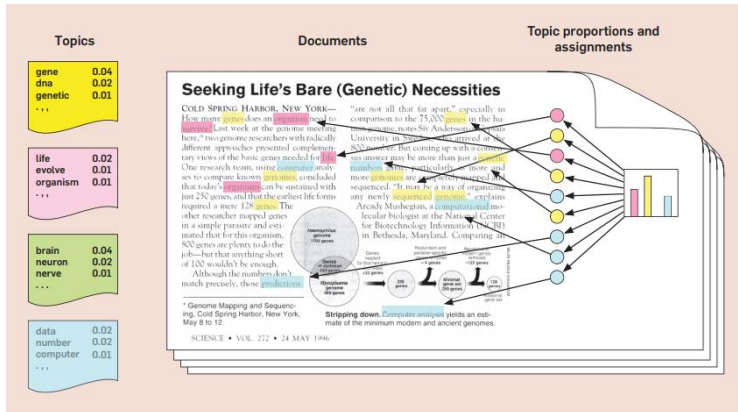
Kesimpulan :

Penelitian ini menggunakan data dari SMS konseling yang dilakukan antara counselor dengan pasiennya. Akurasi dari model bertambah jika persentase percakapan yang dapat dilihat oleh model meningkat. Model dapat membedakan percakapan positif dan negative setelah melihat 20% dari percakapan awal.

2.2 Dasar Teori

2.2.1 Topic Modeling

Konsep *topic modeling* menurut David M. Blei merupakan rangkaian algoritma pembelajaran mesin yang dibuat untuk menemukan probabilitas topik dari arsip dokumen yang berukuran besar dengan informasi tematik. Algoritma *topic modeling* adalah metode statistik yang menganalisis kata-kata dari teks asli untuk menemukan tema sesuai dokumen yang dikumpulkan, bagaimana tema-tema itu saling terhubung satu sama lain, dan bagaimana perubahannya dari waktu ke waktu [4]. Konsep *topic modeling* menurut Blei, ditunjukkan pada Gambar 2.1.



Gambar 2. 1. Konsep *Topic Modeling* menurut Blei [4].

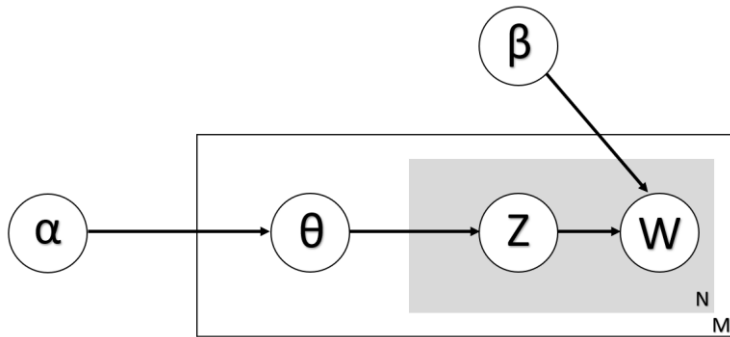
Pendapat lainnya disampaikan oleh Megan R. Brett yang menyatakan bahwa *topic modeling* merupakan bentuk penggalan teks yang dilakukan untuk mengidentifikasi pola corpus. Kemudian algoritma pemrograman digunakan untuk mengambil corpus dan dikelompokkan menjadi kata-kata yang selanjutnya dijadikan topik. Sehingga dapat dikatakan bahwa topik merupakan pola berulang kata-kata yang muncul bersama pada setiap dokumen [5].

Algoritma *topic modeling* tidak memerlukan informasi subjek dan label dengan topik atau kata kunci pada artikelnya. Hal ini dikarenakan distribusi topik dapat ditafsirkan muncul dengan menghitung struktur tersembunyi yang mungkin menghasilkan kumpulan dokumen yang diamati. Struktur tersembunyi yang dapat ditafsirkan ini menganotasikan setiap dokumen dalam koleksi dan anotasi ini dapat digunakan untuk membantu tugas seperti pengambilan informasi, klasifikasi, dan eksplorasi corpus [4].

2.2.2 *Latent Dirichlet Allocation (LDA)*

Latent Dirichlet Allocation merupakan model probabilistik generatif dari korpus. Ide dasar LDA adalah setiap dokumen diwakili sebagai campuran acak pada topik yang tersembunyi, di mana setiap topik dicirikan oleh distribusi

melalui kata-kata yang ada didalamnya [2]. Blei merepresentasikan metode LDA sebagai model probabilistic secara visual seperti pada Gambar 2.2.



Gambar 2. 2. Visualisasi LDA menurut Blei [5]

Berdasarkan visualisasi model di atas, terdapat tiga tingkat pada representasi LDA. Parameter α dan β adalah parameter tingkat corpus yang diasumsikan sebagai sampel sekali dalam proses menghasilkan korpus. Variabel θ adalah variabel tingkat dokumen (M). Variabel z dan w adalah variabel tingkat kata (N) dan dijadikan sampel sekali untuk setiap kata dalam suatu dokumen. Parameter α digunakan dalam menentukan distribusi topik dalam dokumen, semakin besar nilai α maka semakin besar campuran topik yang ada pada suatu dokumen. Sebaliknya, semakin besar kecil nilai α maka semakin kecil campuran topik yang ada pada suatu dokumen. Parameter β digunakan dalam menentukan distribusi kata dalam topik, semakin besar nilai β maka semakin banyak kata-kata yang ada dalam suatu topik. Sebaliknya, semakin kecil nilai β maka semakin kecil sedikit kata-kata yang ada dalam suatu topik. Variabel ini merepresentasikan distribusi topik pada suatu dokumen, semakin besar nilai θ maka semakin banyak topik yang ada pada dokumen tersebut. Sebaliknya, semakin kecil nilai θ maka semakin sedikit topik yang ada pada dokumen tersebut [2].

Secara umum, cara kerja LDA adalah dengan memasukkan kumpulan dokumen dan beberapa parameter yang

ditentukan. Kemudian dilakukan proses LDA sehingga menghasilkan model yang terdiri dari bobot yang dapat dinormalisasi terhadap probabilitas. Probabilitas muncul dalam 2 jenis : (a) probabilitas bahwa dokumen tertentu menghasilkan topik tertentu pada suatu posisi dan (b) probabilitas bahwa topik tertentu menghasilkan kata tertentu dari kumpulan perbendaharaan kata [6].

2.2.3 *TF-IDF*

TF-IDF (*Term Frequency-Inverse Document Frequency*) merupakan perhitungan statistic yang dilakukan untuk mengetahui tingkat pengaruh atau kepentingan sebuah kata pada kumpulan dokumen atau *corpus* [7]. Perhitungan ini sering digunakan sebagai faktor pembobotan dalam pencarian informasi, pengembangan teks, dan pemodelan topik.

TF-IDF bekerja dengan menentukan frekuensi relative kata-kata dalam spesifik dokumen dibandingkan dengan proporsi terbalik dari kata tersebut di seluruh dokumen *corpus*, sehingga perhitungan ini menentukan seberapa relevan kata yang diberikan pada dokumen tertentu [8].

2.2.4 *Gaussian Latent Dirichlet Allocation (Gaussian LDA)*

Pemodelan dengan metode *Latent Dirichlet Allocation* mengasumsikan kosa kata yang tetap dari jenis kata, sehingga tidak dapat menangani *out of vocabulary* (OOV) pada dokumen yang ditangani. Oleh karena itu, diusulkan pendekatan untuk mengatasi masalah tersebut dengan menggambarkan topik dari *Dirichlet Process* dengan distribusi dasar menggunakan kata-kata.

Gaussian LDA mengasumsikan dokumen sebagai kumpulan kata yang saling terhubung dengan topik itu

sendiri sebagai *multivariate Gaussian distributions* pada *embedding space*. *Gaussian LDA* mengkodekan distribusi *Gaussian* untuk menangkap sentralitas dalam *embedding space* dan kata-kata yang terhubung pada lokasi sentral ruang tersebut.

Dengan memanfaatkan kedekatan kata-kata semantik serupa pada *embedding space*, dapat ditetapkan probabilitas topik tinggi untuk kata-kata yang mirip. *Word embeddings* juga telah terbukti untuk menangkap keteraturan *lexico-semantic* dalam bahasa, dimana kata-kata dengan *syntactic* dan *semantic property* yang serupa dapat ditemukan dekat satu sama lain pada *embedding space* [3].

2.2.5 Validasi Topik dengan *Perplexity* dan *Topic Coherence*

Topik model belajar kumpulan kata-kata yang berhubungan dari corpora tanpa pengawasan apapun. Berdasarkan kata-kata yang digunakan dalam dokumen, *topic model* menggali hubungan tingkat topik dengan mengasumsikan bahwa satu dokumen mencakup satu set kecil topik ringkas [9].

Pilihan jumlah topik dapat mempengaruhi interpretability hasil. Solusi dengan terlalu sedikit topik umumnya akan menghasilkan topik yang sangat luas, sementara solusi dengan terlalu banyak topik akan menghasilkan topik yang tidak dapat diinterpretasikan yang memilih kombinasi kata istimewa [10].

Dalam penelitian ini, akan diterapkan dua metode validasi topik, yaitu *Perplexity* dan *Topic Coherence*.

Perplexity adalah pengujian yang dilakukan untuk menilai seberapa baik model yang telah dihasilkan dengan pemodelan LDA. *Perplexity* mengukur seberapa baik model dapat melakukan generalisasi terhadap dokumen [10]. Suatu

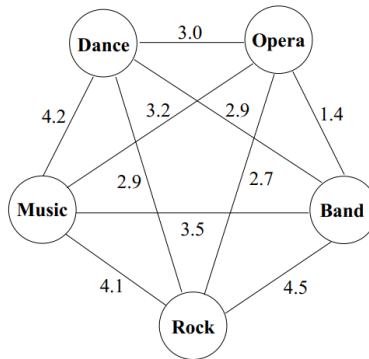
model dikatakan baik jika memiliki nilai *perplexity* yang kecil. Semakin kecil nilai *perplexity*, maka semakin bervariasi dan berbeda topik yang ditemukan oleh model. Sebaliknya, semakin besar nilai *perplexity*, maka semakin sedikit variasi dan perbedaan topik yang ditemukan oleh model.

Topic Coherence merupakan pengujian untuk mengetahui seberapa mudah topik yang dihasilkan oleh model dapat dipahami atau diinterpretasikan oleh manusia [11]. *Topic Coherence* mengukur skor satu topik dengan mengukur tingkat kesamaan semantik antara kata-kata dengan nilai tinggi dalam topik. Langkah-langkah ini membantu membedakan topik yang topik dan topik semantik yang dapat ditafsirkan secara semantik dan interpretif [9].

Dalam penelitian ini, pengujian nilai *perplexity* dan *topic coherence* dari model akan dilakukan dengan menggunakan library yang disediakan oleh python yaitu gensim.

2.2.6 *Pointwise Mutual Information (PMI)*

Pointwise Mutual Information merupakan pengukuran asosiasi dalam teori informasi dan statistik [12] untuk mengetahui hubungan antar *single event*. *Pointwise Mutual Information* dalam penelitian ini digunakan untuk mengetahui hubungan antar kata dalam topik berdasarkan jumlah kemunculan kata serta jumlah co-occurrence masing-masing kata dalam topik [13]. Ilustrasi *Pointwise Mutual Information* ditampilkan pada gambar Gambar 2.3.



Gambar 2. 3. Ilustrasi *Pointwise Mutual Information*

2.2.7 Riliv

Riliv merupakan platform konseling online yang menghubungkan orang-orang dengan masalah pribadi kepada psikolog, melalui SMS, panggilan suara atau video call. Riliv memiliki misi untuk membuat konseling yang mudah dan terjangkau untuk meningkatkan kesejahteraan masyarakat. Riliv dapat mencapai tempat di mana tidak ada psikolog, dan pengguna dapat mengobrol dengan psikolog kapanpun dan dimanapun mereka membutuhkannya dan ini sangat rahasia. Masalah seperti stigma yang menahan orang untuk beralih ke psikolog, perjalanan yang tidak nyaman, dan ketidaknyamanan pribadi pada beberapa orang dalam terapi tatap muka dapat diatasi oleh Riliv. Semua psikolog Riliv juga memiliki pengalaman dan mendapat izin dari Asosiasi Psikolog Indonesia serta memiliki latar belakang yang luas dalam menangani depresi terhadap masalah perkawinan.

Riliv mempermudah konseling dengan membuat sesi terapi di tempat dimana klien merasa nyaman. Klien dapat memilih mengirim SMS, menelepon atau melakukan video call. Konseling in-office bekerja lebih baik untuk beberapa orang, namun konseling online adalah pengobatan yang hamper sama

dan bekerja untuk sebagian besar klien. Selain itu, banyak biaya yang telah dihemat seperti office atau sewa rumah sakit.

Riliv juga telah memenangkan beberapa penghargaan dalam perjalanannya seperti Best Impactful Android Application dari Google Japan 2015 dan 16 Best Young Social Entrepreneurs SEA oleh Singapore International Foundation 2016 dan juga penerima Australian Awards 2017 [1].

2.2.8 Laravel Framework

Web Framework adalah kerangka kerja perangkat lunak yang dirancang untuk mendukung pengembangan aplikasi web termasuk layanan web, sumber web, dan API web. Kerangka kerja web bertujuan untuk mengurangi biaya overhead yang terkait dengan aktivitas umum yang dilakukan dalam pengembangan web.

Laravel merupakan salah satu *web framework open source* yang sangat populer sebagai kerangka aplikasi web yang mudah dan cepat. Taylor Otwell menciptakan Laravel sebagai upaya untuk memberikan alternative kerangka CodeIgniter yang lebih maju, yang tidak menyediakan fitur tertentu seperti dukungan internal untuk authentication dan authorization.

Laravel dibuat dengan kerangka MVC (*Model View Controller*) yang membuatnya lebih berguna daripada PHP. Laravel mencoba untuk menghilangkan kesulitan dalam pembuatan website dengan mengurangi tugas umum yang digunakan di sebagian besar website, seperti authentication, routing, sessions, dan caching. Laravel memiliki arsitektur yang unik, dimana pengembang dimungkinkan untuk membuat infrastruktur pengembang sendiri yang dirancang khusus untuk aplikasi pengembang.

Laravel memiliki beberapa fitur utama, yaitu :

1. *Modularity*. Modularitas merupakan kemampuan sejauh mana komponen system dapat dipisahkan dan digabungkan kembali.

2. *Authentication*. Otentikasi adalah bagian terpenting dari aplikasi website. Pengembang biasanya menghabiskan banyak waktu untuk membuat kode otentikasi. Otentikasi telah dibuat menjadi lebih sederhana di Laravel 5.
3. *Application Logic*. Logika aplikasi dapat diimplementasikan dalam aplikasi apapun baik dengan menggunakan pengendali atau langsung menggunakan syntax yang mirip dengan Sinatra framework. Laravel dirancang dengan hak istimewa yang memberikan pengembang fleksibilitas yang dibutuhkan untuk menciptakan segala sesuatu mulai dari situs yang sangat kecil hingga aplikasi perusahaan yang masif.
4. *Caching*. Caching adalah penyimpanan data sementara yang digunakan untuk menyimpan data sementara dan dapat diambil dengan cepat. Caching sering digunakan untuk mengurangi waktu yang dibutuhkan dalam mengakses database atau layanan jarak jauh lainnya, serta menjadi alat yang baik untuk menjaga aplikasi agar cepat dan reponsif.
5. *Method* atau *Dependency Injection*. Dalam Laravel, *Inversion of Control (IoC)* adalah alat yang ampuh untuk mengelola dependensi kelas. *Dependency Injection* adalah metode untuk menghapus dependensi kelas. Wadah IoC Laravel adalah salah satu fitur Laravel yang paling banyak digunakan.
6. *Routing*. Laraver memberi kemudahan dalam melakukan routing. Rute dapat dipicu dalam aplikasi dengan fleksibilitas dan control yang baik agar sesuai dengan URL.
7. *Restful Controllers*. Kontroler yang tenang memberikan cara opsional untuk memisahkan logika di balik permintaan HTTP GET dan POST.
8. *Testing & Debuging*. Laravel dibangun dengan pengujian dengan dukungan PHPUnit yang sudah disiapkan untuk aplikasi.
9. *Automatic Pagination*. Penyederhanaan tugas pelaksanaan pagination, menggantikan pendekatan manual yang biasa

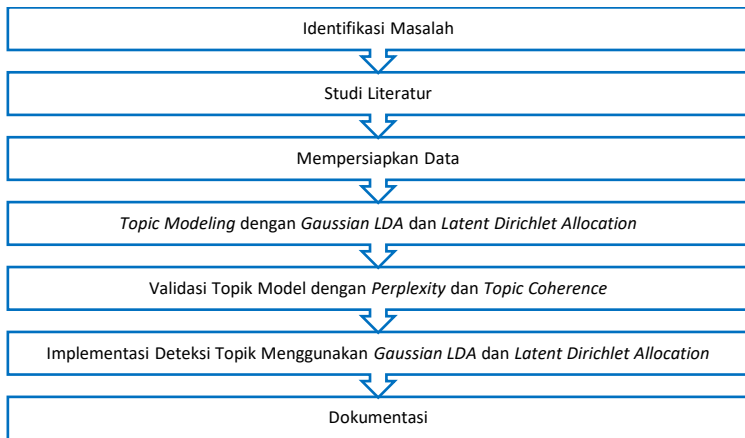
dengan metode otomatis yang diintegrasikan ke dalam Laravel.

10. *Template Engine*. Blade adalah mesin templating sederhana namun hebat yang disediakan pada Laravel. Tidak seperti layout pengendali, blade digerakkan oleh turunan dan bagian template. Semua template blade harus menggunakan ekstensi `.blade.php`.
11. *Database Query Builder*. Pembangunan query database Laravel menyediakan antarmuka yang mudah dan lancar untuk membuat dan menjalankan query database.
12. *Multiple File System*. Laravel 5 mendukung beberapa file system. Laravel menggunakan pihak ketiga Flysystem untuk menyediakan beberapa dukungan file [14].

BAB III METODOLOGI

Pada bab ini akan diuraikan metodologi yang akan digunakan untuk menyusun tugas akhir agar terarah dan sistematis.

3.1 Tahapan Pelaksanaan Tugas Akhir



Gambar 3. 1. Tahapan Pelaksanaan Tugas Akhir

3.1.1 Identifikasi Masalah

Sebagai tahap pertama, tahap identifikasi masalah dilakukan dengan cara observasi pada aplikasi “Riliv - Curhat dengan Psikolog”. Hasil observasi menunjukkan bahwa permasalahan terdapat pada jumlah konseling pengguna yang banyak, dengan topik yang bermacam-macam. Adapun topik yang telah diidentifikasi masih belum mencakup seluruh topik permasalahan pada pengguna, sehingga pengalihan informasi dalam tahapan konseling dengan psikolog dirasa belum optimal. Padahal identifikasi topik sangat diperlukan untuk mengetahui proses konseling yang tepat sesuai permasalahan yang dialami pengguna. Berdasarkan kondisi tersebut, dirasa perlu untuk melakukan identifikasi topik permasalahan apa saja

yang sering dialami pengguna konseling aplikasi “Riliv - Curhat dengan Psikolog”.

3.1.2 Studi Literatur

Tahap studi literatur dilakukan dengan tujuan memahami konsep, metode, dan teknologi sesuai bahasan dan permasalahan sehingga dapat memberi solusi mengenai permasalahan yang akan digunakan dalam penyusunan tugas akhir. Tahap studi literature dilakukan dengan menggali informasi sesuai benang merah penelitian melalui literatur-literatur sebagai sumber terkait konsep-konsep atau penelitian sebelumnya yang pernah dilakukan yang terkait dengan permasalahan dalam bentuk jurnal, buku, maupun referensi online. Adapun literatur utama pada pembahasan *Gaussian LDA* mengacu pada penilitan Rajarshi Das dkk dengan judul *Gaussian LDA for Topic Models with Word Embedding*, pembahasan *topic modeling* dan LDA mengacu pada penelitian David M. Blei, dkk dengan judul *Latent Dirichlet Allocation*, serta pembahasan *Topic Coherence* mengacu pada penelitian Jonathan Chang, dkk dengan judul *Reading Tea Leaves : How Humans Interpret Topic Models*.

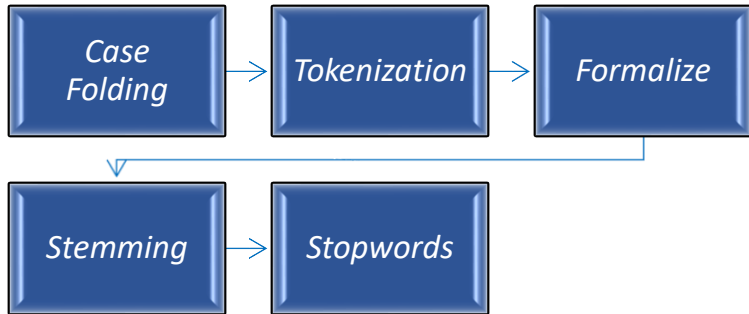
3.1.3 Mempersiapkan Data

Tahap mempersiapkan data terdiri dari sub aktifitas pengumpulan, pemahaman, dan pemilihan data. Tahap ini bertujuan untuk mempersiapkan data yang akan dianalisis menggunakan *topic modeling*. Adapun data yang akan dianalisis merupakan data konseling dari Riliv. Pengumpulan data dilakukan dengan mengambil data konseling Riliv selama bulan januari hingga desember 2017, karena aplikasi Riliv dianggap telah stabil pada waktu ini sehingga minim perbaikan dan digunakan untuk konseling secara maksimal.

3.1.4 Pra-proses Data

Pada tahap ini dilakukan langkah-langkah yang diperlukan sebelum pemodelan topik agar data yang digunakan

sesuai dan siap diolah. Adapun sub-aktivitas dari tahap pra-pemrosesan corpus digambarkan dengan Gambar 3.2.



Gambar 3. 2 Sub-Aktivitas dari Tahap Pra-Pemrosesan Corpus

a. Case Folding

Dalam tahap ini seluruh data diubah menjadi huruf kecil. Hal ini bertujuan agar kata yang sama namun berbeda secara penulisan huruf kapital dan tidak, tidak dianggap kata yang berbeda. Contoh data sebelum dan sesudah melalui tahap case folding ditampilkan pada Tabel 3.1.

Tabel 3. 1. Contoh Data Sebelum dan Sesudah Melalui Tahap *Case Folding*

Sebelum <i>Case Folding</i>	Setelah <i>Case Folding</i>
Salam. Kami (saya dan pasangan) sudah bersama kurang lebih 8 tahun dan daripada pacar sudah saya anggap seperti keluarga sendiri. Saya cenderung termasuk dalam kepribadian Tipe B, sedangkan pasangan saya ke arah Tipe A. Saya	salam. kami (saya dan pasangan) sudah bersama kurang lebih 8 tahun dan daripada pacar sudah saya anggap seperti keluarga sendiri. saya cenderung termasuk dalam kepribadian tipe b, sedangkan pasangan saya ke arah tipe a. saya

<p>sangat terbuka dan sangat jujur dengannya perihal keluarga, teman, maupun jadwal kegiatan sehari-hari. Namun sayangnya ini tidak berbanding lurus dengannya yang sangat menjaga privasi dan memberikan batasan-batasan pada area-area tertentu. Bahkan sekarang saya rasa topiknya (area) semakin bertambah dan berubah menjadi lebih sensitif. Saya sering menemukan gestur-gestur yang (mungkin saya salah) cukup membuat saya mulai khawatir dengan hubungan kami kedepan, seperti ancaman dan makian. Maka untuk saat ini saya hanya dapat menjaga privasinya dan mengalihkan perhatian saya ke tempat lain (bukan pria lain) dan berharap tidak menggangu. Apakah hubungan ini masih wajar? Berusaha optimis. Mohon pencerahannya. Terima kasih, ya.</p>	<p>sangat terbuka dan sangat jujur dengannya perihal keluarga, teman, maupun jadwal kegiatan sehari-hari. namun sayangnya ini tidak berbanding lurus dengannya yang sangat menjaga privasi dan memberikan batasan-batasan pada area-area tertentu. bahkan sekarang saya rasa topiknya (area) semakin bertambah dan berubah menjadi lebih sensitif. saya sering menemukan gestur-gestur yang (mungkin saya salah) cukup membuat saya mulai khawatir dengan hubungan kami kedepan, seperti ancaman dan makian. maka untuk saat ini saya hanya dapat menjaga privasinya dan mengalihkan perhatian saya ke tempat lain (bukan pria lain) dan berharap tidak menggangu. apakah hubungan ini masih wajar? berusaha optimis. mohon pencerahannya. terima kasih, ya.</p>
--	--

b. *Tokenization*

Tokenization adalah proses memisahkan kumpulan kata dalam kalimat menjadi potongan kata tunggal. Proses tokenization dilakukan untuk mempersiapkan dokumen pada proses berikutnya, yaitu stopwords. Contoh data sebelum dan sesudah melalui tahap tokenization ditampilkan pada Tabel 3.2.

**Tabel 3. 2. Contoh Data Sebelum dan Sesudah Melalui Tahap
Tokenization**

Sebelum Tokenization	Setelah Tokenization
<p>betul dari dulu saya damba salah satu institut negeri daerah yogyakarta tahun kemarin pun saya coba ikut sampai banyak hal yang saya lalu saya rasa sekali juang demi saya bisa masuk ke sana tapi allah belum hendak saya tidak lulus tiga kali itu kecewa itu pasti saya sempat juga bingung akhir sekarang saya kuliah di putus betul saya anak akhir ayah saya sudah tinggal harap ibu saya hanya pada saya karena kakak saya sudah meni semua saya kuliah juga dari uang tinggal ayah saya hati saya tidak bisa bohong kalau saya masih penasaran dengan institut sebut saya masing ingin coba karena memang pada dasar institut sebut banyak minat dan rata minat pun rela kali tes sampai dapat sana nah turut anda salah tidak saya coba lagi apa saya seperti kurang syukur</p>	<p>[`betul`, `dari`, `dulu`, `saya`, `damba`, `salah`, `satu`, `institut`, `negeri`, `daerah`, `yogyakarta`, `tahun`, `kemarin`, `pun`, `saya`, `coba`, `ikut`, `sampai`, `banyak`, `hal`, `yang`, `saya`, `lalu`, `saya`, `rasa`, `sekali`, `juang`, `demi`, `saya`, `bisa`, `masuk`, `ke`, `sana`, `tapi`, `allah`, `belum`, `hendak`, `saya`, `tidak`, `lulus`, `tiga`, `kali`, `itu`, `kecewa`, `itu`, `pasti`, `saya`, `sempat`, `juga`, `bingung`, `akhir`, `sekarang`, `saya`, `kuliah`, `di`, `putus`, `betul`, `saya`, `anak`, `akhir`, `ayah`, `saya`, `sudah`, `tinggal`, `harap`, `ibu`, `saya`, `hanya`, `pada`, `saya`, `karena`, `kakak`, `saya`, `sudah`, `meni`, `semua`, `saya`, `kuliah`, `juga`, `dari`, `uang`, `tinggal`, `ayah`, `saya`, `hati`, `saya`, `tidak`, `bisa`, `bohong`, `kalau`, `saya`, `masih`, `penasaran`, `dengan`, `institut`, `sebut`, `saya`, `masing`, `ingin`, `coba`,</p>

	`karena`, `memang`, `pada`, `dasar`, `institut`, `sebut`, `banyak`, `minat`, `dan`, `rata`, `minat`, `pun`, `rela`, `kali`, `tes`, `sampai`, `dapat`, `sana`, `nah`, `turut`, `anda`, `salah`, `tidak`, `saya`, `coba`, `lagi`, `apa`, `saya`, `seperti`, `kurang`, `syukur`]
--	--

c. *Formalize*

Formalize adalah proses mengubah kata yang dianggap salah dengan kata pembenarannya yang memiliki makna. Proses *formalize* bertujuan untuk memperbaiki kata typo atau salah penulisan dan kata yang dianggap belum baku. Proses ini dilakukan dengan memanfaatkan kamus kata baku Bahasa Indonesia dari Kateglo dan kamus kata *formalizer* yang telah disusun Dr. Eng. Ayu Purwarianti, ST.,MT., et al [15]. Contoh data sebelum dan sesudah melalui tahap *formalize* ditampilkan pada Tabel 3.3.

Tabel 3. 3. Contoh Data Sebelum dan Sesudah Melalui Tahap *Formalize*

Sebelum <i>Formalize</i>	Setelah <i>Formalize</i>
kebetulan dari dulu sy mendambakan salah satu institut negeri daerah jogja thn kemarin pun sy coba ikut sampai 3x, banyak hal yg sy lalui sy merasakan sekali perjuangan demi sy bisa masuk kesana tp allah blm berkehendak sy tidak lulus tiga kali itu. kecewa itu pasti sy sempat down jg bingung, akhirnya skrng sy kuliah di	kebetulan dari dulu saya mendambakan salah satu institut negeri daerah yogyakarta tahun kemarin pun saya coba ikut sampai banyak hal yang saya lalui saya merasakan sekali perjuangan demi saya bisa masuk ke sana tapi allah belum berkehendak saya tidak lulus tiga kali itu kecewa itu pasti saya sempat

<p>pts, kebetulan sy anak terakhir ayah sy sdh meninggal harapan ibu sy hanya pd sy karna kaka sy sdh menikah semua, sy kuliah jg dr uang peninggalan ayah sy.</p> <p>hati sy tdk bisah dibohongi kalau sy masih penasaran dengan institut tersebut sy masing ingin mencoba, karna memang pada dasarnya institut tersebut bnyk peminat dan rata-rata peminatnya pun rela berkali-kali tes sampai dapat disana.</p> <p>nah menurut anda salah tidak sy mencoba lagi?apa sy seperti kurang bersyukur?</p>	<p>juga bingung akhirnya sekarang saya kuliah di putus kebetulan saya anak terakhir ayah saya sudah meninggal harapan ibu saya hanya pada saya karena kakak saya sudah menikah semua saya kuliah juga dari uang peninggalan ayah saya hati saya tidak bisa dibohongi kalau saya masih penasaran dengan institut tersebut saya masing ingin mencoba karena memang pada dasarnya institut tersebut banyak peminat dan rata-rata peminatnya pun rela berkali-kali tes sampai dapat disana nah menurut anda salah tidak saya mencoba lagi apa saya seperti kurang bersyukur</p>
---	---

d. *Stemming*

Stemming merupakan proses mengubah suatu kata menjadi kata dasar dari kata tersebut. *Stemming* bekerja dengan menghilangkan semua imbuhan (*affixes*), baik yang terdiri dari awalan (*prefixes*), sisipan (*infixes*), akhiran (*suffixes*), dan kombinasi dari awalan dan akhiran (*confixes*) pada kata turunan. *Stemming* dilakukan agar tidak ada kata yang sama namun dianggap berbeda karena adanya imbuhan (*affixes*). Proses *stemming* dilakukan dengan menggunakan library Sastrawi, yaitu library stemmer bahasa indonesia dengan lisensi MIT. Contoh data sebelum dan sesudah melalui tahap *stemming* ditampilkan pada Tabel 3.4.

Tabel 3. 4. Contoh Data Sebelum dan Sesudah Melalui Tahap *Stemming*

Sebelum <i>Stemming</i>	Setelah <i>Stemming</i>
kebetulan dari dulu saya mendambakan salah satu institut negeri daerah yogyakarta tahun kemarin pun saya coba ikut sampai banyak hal yang saya lalui saya merasakan sekali perjuangan demi saya bisa masuk ke sana tapi allah belum berkehendak saya tidak lulus tiga kali itu kecewa itu pasti saya sempat juga bingung akhirnya sekarang saya kuliah di putus kebetulan saya anak terakhir ayah saya sudah meninggal harapan ibu saya hanya pada saya karena kakak saya sudah menikah semua saya kuliah juga dari uang peninggalan ayah saya hati saya tidak bisa dibohongi kalau saya masih penasaran dengan institut tersebut saya masing ingin mencoba karena memang pada dasarnya institut tersebut banyak peminat dan rata-rata peminatnya pun rela berkali-kali tes sampai dapat disana nah menurut anda salah tidak	betul dari dulu saya damba salah satu institut negeri daerah yogyakarta tahun kemarin pun saya coba ikut sampai banyak hal yang saya lalui saya rasa sekali juang demi saya bisa masuk ke sana tapi allah belum hendak saya tidak lulus tiga kali itu kecewa itu pasti saya sempat juga bingung akhir sekarang saya kuliah di putus betul saya anak akhir ayah saya sudah tinggal harap ibu saya hanya pada saya karena kakak saya sudah meni semua saya kuliah juga dari uang tinggal ayah saya hati saya tidak bisa bohong kalau saya masih penasaran dengan institut sebut saya masing ingin coba karena memang pada dasar institut sebut banyak minat dan rata minat pun rela kali tes sampai dapat sana nah turut anda salah tidak saya coba lagi apa saya seperti kurang syukur

saya mencoba lagi apa saya seperti kurang bersyukur	
--	--

e. **Stopword Removal**

Stopwords merupakan kata umum yang biasanya muncul dalam jumlah besar dan dianggap tidak memiliki makna. Adapun daftar *stopwords* yang digunakan adalah stopwords Bahasa Indonesia yang disusun berdasarkan penelitian Dr. Eng. Ayu Purwarianti, ST.,MT. dkk. Beberapa contoh stopwords yang tersimpan dalam daftar yang dimaksud adalah sebagai berikut :

Tabel 3. 5. Contoh Data *Stopwords*

Daftar <i>Stopwords</i>
`artinya`, `asal`, `asalkan`, `atas`, `atau`, `ataukah`, `ataupun`, `awal`, `awalnya`, `bagai`, `bagaikan`, `bagaimana`, `bagaimanakah`, `bagaimanapun`, `buat`, ``, `bukan`, `bukankah`, `bukanlah`, `bukannya`, `bulan`, `bung`, `cara`, `caranya`, `cukup`, `cukupkah`, `cukuplah`, `cuma`, `dahulu`, `dalam`, `dan`, `dapat`, `dari`, `daripada`, `datang`, `dekat`

Menghilangkan *stopwords* merupakan tahap yang penting, mengingat banyaknya frekuensi kemunculan stopwords dalam dokumen, yang beujung pada tingginya probabilitas kata stopwords dalam topik dan menyebabkan tpik tidak dapat diinterpretasikan dengan baik. Contoh data sebelum dan sesudah melalui penghapusan *stopwords* ditampilkan pada Tabel 3.6.

Tabel 3. 6. Contoh Data Sebelum dan Sesudah Melalui Tahap *Stopwords*

Sebelum <i>Stopwords</i>	Setelah <i>Stopwords</i>
betul, dari, dulu, saya, damba, salah, satu, institut, negeri, daerah, yogyakarta,	damba, institut, negeri, daerah, yogyakarta, kemarin, coba, juang,

<p>tahun, kemarin, pun, saya, coba, ikut, sampai, x, banyak, hal, yang, saya, lalu, saya, rasa, sekali, juang, demi, saya, bisa, masuk, ke, sana, tapi, allah, belum, hendak, saya, tidak, lulus, tiga, kali, itu, kecewa, itu, pasti, saya, sempat, down, juga, bingung, akhir, sekarang, saya, kuliah, di, putus, betul, saya, anak, akhir, ayah, saya, sudah, tinggal, harap, ibu, saya, hanya, pada, saya, karena, kakak, saya, sudah, menikah, semua, saya, kuliah, juga, dari, uang, tinggal, ayah, saya, hati, saya, tidak, bisa, bohong, kalau, saya, masih, penasaran, dengan, institut, sebut, saya, masing, ingin, coba, karena, memang, pada, dasar, institut, sebut, banyak, minat, dan, rata, minat, pun, rela, kali, tes, sampai, dapat, sana, nah, turut, anda, salah, tidak, saya, coba, lagi, apa, saya, seperti, kurang, syukur</p>	<p>masuk, allah, lulus, kali, kecewa, down, bingung, kuliah, putus, anak, ayah, tinggal, harap, kakak, menikah, kuliah, uang, tinggal, ayah, hati, bohong, penasaran, institut, coba, dasar, institut, minat, minat, rela, kali, tes, coba, syukur</p>
--	--

3.1.5 Pemodelan Topik

Tahap pemodelan topik bertujuan untuk menghasilkan model topik yang paling tepat untuk data. Untuk menghasilkan model yang tepat, pengujian model topik dilakukan secara berulang dengan mengubah nilai input parameter. Pada penelitian ini, pemodelan topik dilakukan metode *Gaussian Latent Dirichlet Allocation (LDA)*. Pada tahap ini juga akan

dilakukan pemodelan topik dengan metode *LDA* sebagai pembandingan metode *Gaussian LDA*.

Untuk mendapatkan model dengan hasil luaran yang paling baik, eksperimen dilakukan dengan menentukan nilai masukkan dari parameter metode *Gaussian LDA* dan *LDA*. Parameter yang dimaksud untuk metode *Gaussian LDA* dan *LDA* adalah jumlah *passes* atau iterasi dan jumlah topik. Model dengan hasil luaran yang paling baik akan dipilih melalui tahap validasi model.

3.1.6 Validasi Model Topik

Validasi model topik metode dilakukan dengan melihat nilai *perplexity* yang paling kecil sesuai banyak uji coba topik dan *passes* yang dilakukan. Sedangkan validasi topik metode dilakukan berdasarkan tingkat koherensi atau tingkat kemudahannya dalam diinterpretasi manusia. Dalam tugas akhir ini, hasil model topik dengan metode *Gaussian LDA* dan *Latent Dirichlet Allocation* akan dibandingkan dengan menggunakan perhitungan *Pointwise Mutual Information*.

Mengingat tahap validasi bertujuan memastikan luaran dari hasil *topic modeling* yang dilakukan agar memiliki nilai yang baik, apabila nilai masih belum optimal maka akan kembali ke tahap pembentukan model topik untuk melakukan eksperimen kembali dengan memasukkan parameter yang baru.

3.1.7 Implementasi Deteksi Topik Menggunakan Hasil Pemodelan Topik

Pada tahap ini dilakukan pembuatan aplikasi deteksi topik menggunakan hasil pemodelan topik. Aplikasi deteksi topik dibuat untuk memudahkan psikolog memahami lebih awal topik dari konseling yang dilakukan oleh pengguna, sehingga konseling dan penanganan masalah dapat dilakukan lebih awal serta lebih terarah. Framework *Laravel* dipilih untuk digunakan dalam melakukan pembuatan aplikasi deteksi topik.

Halaman Sengaja Dikosongkan

BAB IV

PERANCANGAN

Pada bab ini akan dijelaskan mengenai perancangan penelitian tugas akhir. Perancangan ini diperlukan sebagai panduan dalam melakukan penelitian tugas akhir, yang dijelaskan sebagai berikut.

4.1 Pengambilan Data

Dalam pelaksanaan identifikasi topik dan implementasi aplikasi deteksi topik konseling pelanggan Riliv, data merupakan objek utama analisis. Data yang dibutuhkan merupakan data tipe teks konseling pelanggan dengan psikolog Riliv.

Pengambilan data dilakukan dari ekstraksi database aplikasi Riliv mulai dari Januari 2017 hingga Desember 2017. Atribut pada basis data konseling Riliv diantaranya adalah Nomor Konseling, ID User, ID Psikolog, Tanggal Konseling, Konten, Rating. Data yang diekstraksi dari basis data adalah sejumlah 113975 data.

4.2 Metodologi Implementasi Penilaian

Metodologi implementasi penelitian merupakan tahapan yang dilakukan untuk mencapai tujuan penelitian yang disesuaikan dengan komputasi secara otomatis. Komputasi yang dilakukan pada penelitian ini adalah dengan beberapa tahapan, yakni pemrosesan data, pembuatan model, serta implementasi deteksi topik menggunakan framework Laravel. Ada 5 tahapan utama dalam melakukan implementasi penelitian yaitu mempersiapkan data, pra-proses data, pemrosesan data, validasi model, dan implementasi aplikasi.

4.2.1 Mempersiapkan Data

Mempersiapkan data merupakan tahap untuk mengolah data yang sudah dimiliki menjadi bentuk struktur data yang dibutuhkan oleh tahap proses data sehingga siap digunakan

untuk melakukan tahap proses data dan analisis. Alir tahap persiapan data sesuai dengan Gambar 4.1.

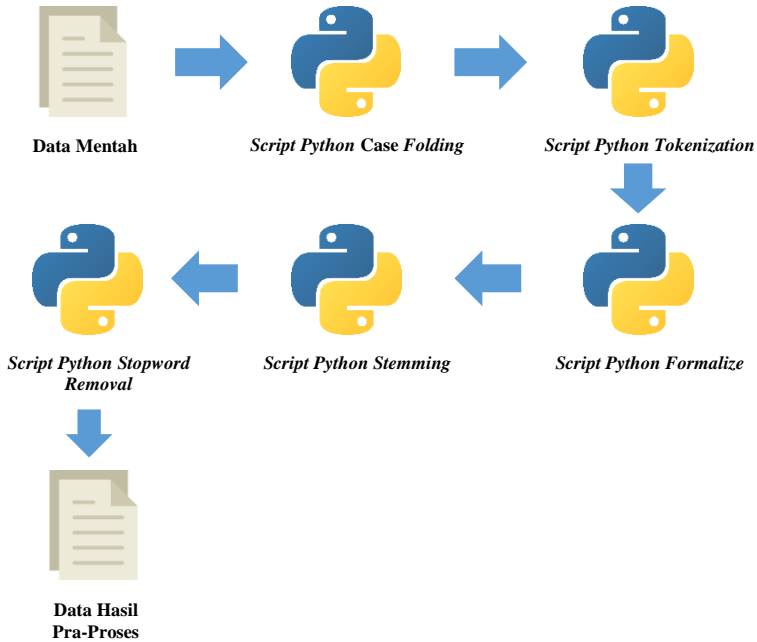


Gambar 4. 1. Alur Mepersiapkan Data

Data mentah dalam format csv kemudian dikonversikan menjadi *file .txt* menggunakan kode program dengan bahasa pemrograman *python* dan data yang diambil adalah data pada kolom konteks konseling. Data selanjutnya dipilah untuk menghilangkan duplikat data dikarenakan aplikasi Rilis menyimpan setiap konseling menjadi 3 buah data yang sama.

4.2.2 Pra-Proses Data

Tahap pra-proses data merupakan tahapan yang dilakukan dalam penelitian penggalian data. Pra-proses data memiliki tujuan untuk menyeragamkan data dan membuang kata-kata yang tidak bermakna. Pada tahap ini dilakukan aktivitas *case folding* atau pengubahan data menjadi huruf kecil, *formalize* atau perbaikan data yang memiliki kesalahan penulisan, *stemming* atau penghapusan imbuhan kata agar kembali menjadi kata dasar, *tokenization* atau pemisahan kata-kata pada sebuah kalimat menjadi data token dan disimpan menjadi variabel array, dan *stopwords removal* atau penghapusan kata-kata yang tidak bermakna informasi. Alur pra-proses data dapat dilihat pada Gambar 4.2.



Gambar 4. 2. Alur Tahap Pra-Proses Data

Langkah pertama yang dilakukan adalah pengubahan kata menjadi huruf kecil atau *case folding*, penghapusan simbol dan angka dari kalimat menggunakan *library RegExr*.

Selanjutnya dilakukan perbaikan data yang memiliki kesalahan penulisan dengan kamus data *FormalizationDict* pada *Indonesian Natural Language Processing REST API* [16] yang disusun oleh Dr. Eng. Ayu Purwarianti, ST.,MT., et al [15]. Penambahan data normalisasi kata ke dalam kamus data *FormalizationDict* juga dilakukan secara manual untuk memperbanyak perbendaharaan kata tidak baku beserta rujukan kata perbaikannya.

Kemudian dilakukan proses penghapusan imbuhan kata dengan menggunakan *library Sastrawi* di bawah lisensi MIT. Langkah terakhir yang dilakukan adalah penghapusan kata-kata

yang tidak bermakna informasi serta penggabungan kata sambung atau frasa dengan menggunakan tanda “_”.

Setelah melalui pra-proses di atas, data dianggap sudah bersih dan siap untuk diproses pada tahap proses data.

4.2.3 Pemodelan Topik dengan Metode *Gaussian LDA*

Pada tahap ini dilakukan pemodelan topik dengan metode *Gaussian Latent Dirichlet Allocation*. Sebelum pemodelan topik, terlebih dahulu dilakukan pemrosesan *corpus* dan pemrosesan vektor kata sebelum dilakukan pemodelan topik.

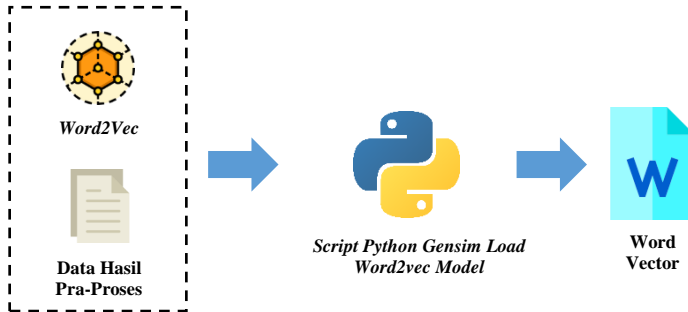
a. Pemrosesan *Corpus*

Pada tahap ini dilakukan pemrosesan pada data hasil pra-proses data. Data masukkan diproses menjadi *corpus* yang merupakan yang merupakan data map berisikan kata dan topik dari setiap dokumen.

Pada tahap ini juga dilakukan penghapusan terhadap data yang tingkat kemunculan terlalu sering atau terlalu jarang pada dokumen. Nilai maksimum dan minimum kemunculan akan diatur secara manual.

b. Pemrosesan Vektor Kata

Pada tahap ini dilakukan pemrosesan kata untuk diubah menjadi bentuk vektor. Pengubahan kata menjadi bentuk vektor dilakukan dengan menggunakan model *word2vec* yang telah disusun oleh Dr. Eng. Ayu Purwarianti, ST.,MT., et al [15]. Alur pemrosesan vektor kata dapat dilihat pada Gambar 4.3.



Gambar 4. 3. Pemrosesan Vektor Kata

c. Proses Pemodelan Topik

Pemodelan topik dilakukan dengan komputasi otomatis menggunakan bahasa python. Dalam membentuk model, dilakukan eksperimen pada jumlah topik dan jumlah iterasi yang sebagai parameter masukkan pemodelan *Gaussian LDA*. Model yang dihasilkan kemudian dievaluasi dengan mengevaluasi nilai *perplexity* untuk menentukan model terbaik. Alur pemodelan topik dapat dilihat ditampilkan pada Gambar 4.4.



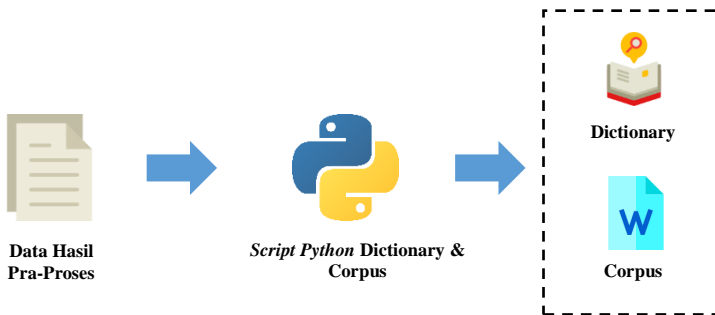
Gambar 4. 4. Proses Pemodelan Topik dengan Metode *Gaussian LDA*

4.2.4 Pemodelan Topik dengan Metode *Latent Dirichlet Allocation*

Pada tahap ini dilakukan pemodelan topik dengan metode *Latent Dirichlet Allocation*. Sebelum pemodelan topik, terlebih dahulu dilakukan pembentukan *dictionary* dan *corpus*.

a. Pembentukan *Dictionary* dan *Corpus*

Pembentukan *dictionary* dan *corpus* dilakukan sebelum tahap *topic modeling* dengan menggunakan data hasil pra-proses. *Dictionary* merupakan format data yang mengandung himpunan kata unik dengan nomor indeks masing-masing. Sedangkan *corpus* merupakan format data berbentuk *bag-of-word reference* yang akan digunakan untuk pembentukan model. Tahapan ini dilakukan sesuai dengan alur pada Gambar 4.5.

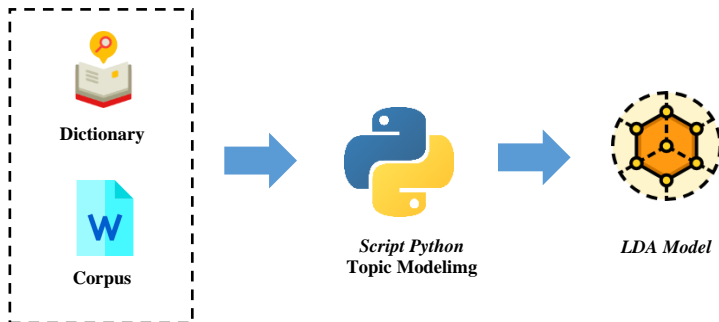


Gambar 4. 5. Alur Pembentukan *Dictionary* dan *Corpus*

Pada tahap ini juga dilakukan penghapusan terhadap data yang tingkat kemunculan terlalu sering atau terlalu jarang pada dokumen. Nilai maksimum dan minimum kemunculan akan diatur secara manual.

b. Proses Pemodelan Topik

Pemodelan topik metode *Latent Dirichlet Allocation* dilakukan dengan menggunakan library *ldamodel* dari *gensim*. Tahapan ini dilakukan sesuai dengan alur pada Gambar 4.6.



Gambar 4. 6. Alur Pembentukan Model LDA

Eksperimen pembentukan model dilakukan dengan menentukan jumlah topik yang menjadi kelompok klasterisasi kata dan penentuan jumlah *passes* dalam *topic modeling* yang merupakan jumlah iterasi proses pembelajaran dari *training model*. Hasil model dengan nilai *perplexity* terkecil dan tidak berubah-ubah akan dipilih menjadi model yang digunakan dan dianggap yang paling mendekati akurat.

4.2.5 Validasi Model Topik

Tahapan ini bertujuan untuk memastikan model yang dibentuk dari proses *topic modeling* pada dokumen yang dihasilkan memiliki nilai probabilitas tertinggi, baik luaran berupa topik maupun kata-kata yang menyusun setiap topiknya. Berikut hal yang diperhatikan dalam tahap validasi model :

1. Jumlah *passes* yang tepat untuk membentuk model.
2. Jumlah topik yang sesuai berdasarkan nilai *perplexity* yang dihasilkan.
3. Jumlah topik yang sesuai berdasarkan nilai *topic coherence* tiap model.

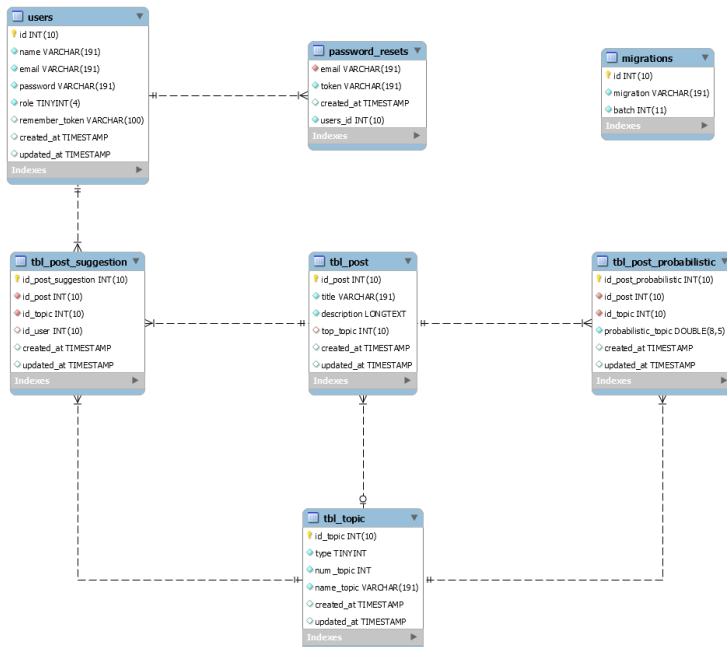
4.2.6 Konstruksi Perangkat Lunak

Berikut ini adalah perancangan perangkat lunak untuk implementasi Aplikasi Deteksi Topik yang selanjutnya dinamakan Aplikasi Portal Reliever, meliputi perancangan

basis data, pembuatan *use case diagram* dan *use case description*.

a. Perancangan Basis Data

Basis data merupakan hal yang penting dalam menyimpan data konseling pelanggan agar dapat diakses oleh aplikasi. Gambar 4.7. menampilkan struktur basis data dari Aplikasi Portal Reliever.



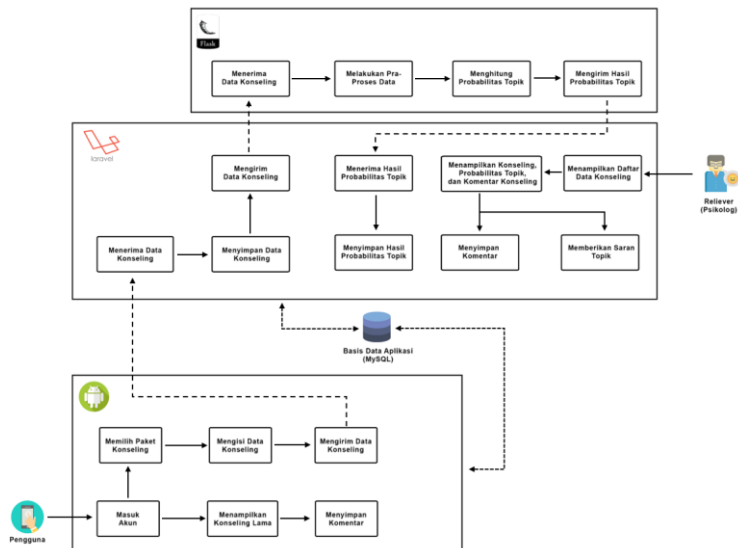
Gambar 4. 7. Perancangan Basis Data Aplikasi Portal Reliever

Dalam perancangan basis data terdapat tabel *users* sebagai penyimpan data akun reliever, *password_resets* sebagai penyimpanan reset password akun, dan *migrations* sebagai penyimpanan riwayat migrations Laravel. Kemudian tabel *tbl_post* sebagai penyimpanan *posting* atau teks konseling dari pengguna, tabel *tbl_topic* sebagai penyimpanan daftar topik, *tbl_post_probabilistic* sebagai penyimpanan perhitungan probabilitas topik untuk setiap *posting*, dan tabel

`tbl_post_suggestion` sebagai penyimpanan saran topik untuk setiap *posting* dari *reliever*. Kolom `top_topic` pada `tbl_post` digunakan untuk menunjukkan topik dengan nilai probabilitas tertinggi serta menjadi topik yang paling relevan terhadap *posting* atau konseling tersebut.

b. Roadmap Perangkat Lunak

Konseling psikolog pada Riliv dilakukan online menggunakan aplikasi android. Aplikasi ini dibuat untuk pengguna dalam menyampaikan permasalahan yang dialami dan melakukan sesi konseling. Aplikasi tersebut juga diperuntukkan kepada psikolog dalam menjawab setiap sesi konseling dengan seluruh pengguna.



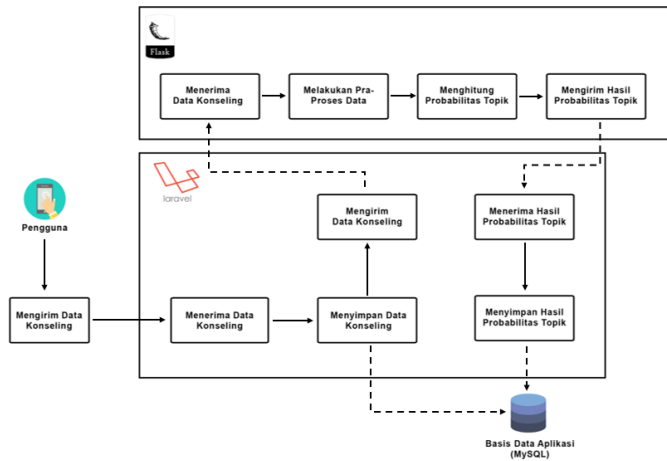
Gambar 4. 8. Roadmap Pengembangan Alur Konseling Online

Dalam perkembangan konseling online, Riliv berusaha untuk memberikan pelayanan yang terbaik untuk pengguna dan psikolog atau reliever. Oleh karenanya, dilakukan pengembangan aplikasi website yang diperuntukkan kepada psikolog. Diharapkan pengembangan aplikasi tersebut dapat

memudahkan dalam melakukan sesi konseling, terutama menyelesaikan permasalahan ketidaknyamanan reliever saat melakukan konseling menggunakan android. Hal ini disebabkan reliever sering mengalami kelelahan dalam mengetik jawaban yang panjang menggunakan android. Gambar 4.8 menampilkan roadmap pengembangan alur konseling online, dimana terdapat aplikasi baru berbasis website yang digunakan reliever dalam konseling. Pengembangan aplikasi website dengan framework Laravel dan framework Flask yang dimanfaatkan untuk mendeteksi probabilitas topik konseling menjadi cakupan dalam penelitian ini.

c. Menghubungkan *Laravel* dan *Flask*

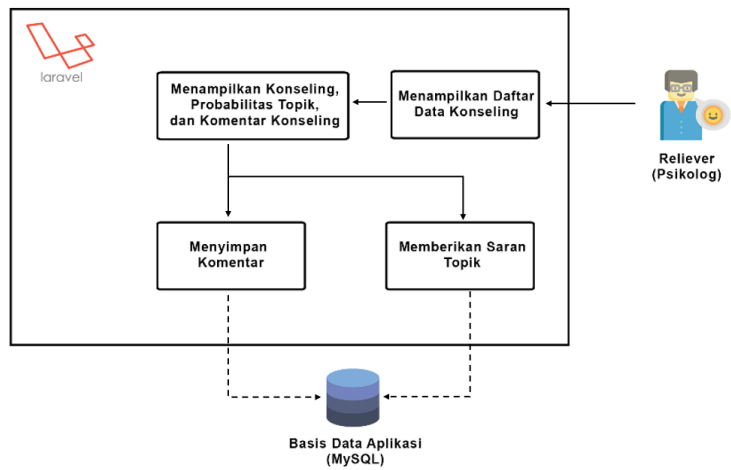
Pada tahapan ini dilakukan penghubungan data dari input melalui kode PHP aplikasi *laravel* dengan kode program pra-proses dan model python pada aplikasi *flask* untuk mendeteksi probabilitas topik dari suatu dokumen baru. Penjelasan secara detail dari rancangan konektivitas aplikasi *laravel* dengan aplikasi *flask* ditampilkan pada Gambar 4.9. Data konseling baru dimasukkan melalui antarmuka aplikasi *laravel*, kemudian data tersebut dikirimkan ke aplikasi *flask* untuk menjalankan model dan menghitung probabilitas topik yang muncul dari data. Hasil dari perhitungan akan disimpan kedalam basis data.



Gambar 4. 9. Alur Penyimpanan *Posting* dan Penghitungan Probabilitas Topik

d. Desain Antarmuka Implementasi Aplikasi

Setelah mendapatkan model yang tepat dan perhitungan probabilitas topik pada dokumen, selanjutnya dilakukan perhitungan presentase probabilitas topik pada dokumen yang diujikan. Hasil perhitungan probabilitas topik pada dokumen kemudian divisualisasikan ke dalam dashboard dengan menggunakan framework *Laravel*. Penjelasan tentang alur penampilan data konseling atau data konseling serta probabilitas topik oleh Reliever ditampilkan pada Gambar 4.10.



Gambar 4. 10. Alur Penampilan Data Konseling dan Probabilitas Topik

BAB V

IMPLEMENTASI

Bab ini menjelaskan hasil dari implementasi perancangan studi kasus atau hasil dari proses pelaksanaan penelitian. Hasil yang akan dijabarkan adalah hasil eksperimen terhadap data yang digunakan sebagai acuan penelitian. Selain itu, akan dijelaskan juga mengenai hambatan dan rintangan dalam proses pelaksanaan penelitian.

5.1 Perangkat Penelitian

Dalam pelaksanaan identifikasi topik dan implementasi aplikasi deteksi topik data konseling pelanggan Riliv, dibutuhkan perangkat untuk menunjang keberlangsungan penelitian dengan spesifikasi sesuai Tabel 5.1.

Tabel 5. 1. Spesifikasi Komputer

<i>Windows Based Operating Systems</i>	
Prosesor	Intel® Core™ i7-7700HQ CPU @2.80GHz
Memory	16GB RAM DDR4 (8GB x 2 Dual Channel)
Graphic Card	Intel HD Graphics 630 – nVidia GeForce GTX1050 4GB DDR5
Sistem Operasi	<i>Windows 10 Education</i>
Arsitektur Sistem	<i>64-bit Operating System, x64-based processor</i>

Model dan aplikasi dikembangkan dengan menggunakan beberapa teknologi seperti editor, bahasa pemrograman, dan *library* yang disajikan dalam Tabel 5.2.

Tabel 5. 2. Teknologi yang Digunakan Untuk Mengembangkan Model

Bahasa Pemrograman	<ul style="list-style-type: none"> • Python 3.6.5 • HTML 5 • CSS 3 • PHP 7.1.9
Editor (IDE)	<ul style="list-style-type: none"> • <i>Notepad++</i> • <i>Visual Studio Code</i> • <i>Sublime Text Editor</i>
Software/Tools	Microsoft Excel 2016
Library	<ul style="list-style-type: none"> • Anaconda 5.1 • Gensim 3.1 • Sastrawi 1.0.1 • NLTK 3.2.5 • RegExr 3.0 • LangDetect 1.0.7 • Pickle • Numpy

5.2 Mempersiapkan Data

Sebelum melakukan *topic modeling*, dilakukan persiapan data untuk untuk diolah sehingga siap digunakan dalam pemrosesan. Data konseling diambil dari basis data aplikasi Riliv dan kemudian di-*export* kedalam dokumen dengan format *.csv*. Pembacaan data dengan format dokumen *.csv* dilakukan dengan menggunakan modul CSV. Kode 5.1 merupakan kode yang digunakan dalam proses memuat data mentah sekaligus menghapus data konseling yang sama. Data yang dihasilkan selanjutnya digunakan sebagai masukan atau bahan untuk melakukan pra-proses data.

```

import csv
with open('post.csv', encoding="utf8") as myFile:
    with open('1_post_noduplicate.txt', 'w',
              encoding="utf8") as outfile:
        reader = csv.reader(myFile)
        line_before = ''
        for row in reader:
            line = row[9]
            if(line != line_before and line != ''
               and line != ' '):
                line = line.replace("\n", "")
                outfile.write(line)
                outfile.write('\n')
            line_before = line

```

Kode 5. 1. Kode Program Memuat Data dan Menghapus Data yang Sama

5.3 Pra-Proses Data

Tahap pra-proses data merupakan tahap yang dilakukan agar data yang digunakan sesuai dengan masukkan yang dibutuhkan dalam proses pembuatan model. Terdapat lima tahapan yang dilakukan dalam pra-proses data, yaitu *case folding*, *tokenization*, *formalize*, *stemming*, dan *stopword removal*.

5.3.1 *Case Folding, Tokenizing, dan Formalize*

Langkah awal sebelum melakukan *case folding*, *tokenizing* dan *formalize* adalah menginisiasi kamus kata baku Bahasa Indonesia dan kamus pengubahan kata yang nantinya akan digunakan untuk pengubahan kata. Kode program yang digunakan sesuai dengan Kode 5.2.

```

import sys
import csv
import nltk
import re

from nltk.tokenize import RegexpTokenizer

from langdetect import detect_langs
from langdetect import DetectorFactory
from langdetect import detect

def initiation_dictionary():
    with open('lib-dict/katabaku.csv', encoding="utf8") as dictionary1:
        for line1 in dictionary1:
            katabaku_dict.append(line1.split('\n')[0])
    with open('lib-dict/formalizationDict.csv', encoding="utf8") as dictionary4:
        for line4 in dictionary4:
            formalizer_dict.append(line4.replace("\n",
""").split(',')')
            formalizer_list.append(line4.replace("\n",
""").split(',')')[0])

#----- Inisiasi variable -----#
katabaku_dict = []
formalizer_dict = []
formalizer_list = []

```

Kode 5. 2. Kode Program Menginisiasi Kamus *Formalize*

Case folding dilakukan dengan menggunakan fungsi *.lower()* yang terdapat pada python dan penghapusan simbol menggunakan *library RegExr*. Kemudian dilakukan *tokenizing* menggunakan *library nltk.tokenize* dengan modul *RegexpTokenizer*. Selanjutnya dilakukan *formalize* dengan memanfaatkan kamus pengubahan data atau *formalizer_dict* yang telah diinisiasi. Pada tahap *formalize* juga dilakukan pengecekan bahasa pada setiap kata dengan menggunakan *library langdetect* dan akan dihapus apabila kata tersebut termasuk kata Bahasa Inggris sesuai hasil *langdetect*. Kode program yang digunakan *case folding*, *tokenizing* dan *formalize* sesuai dengan Kode 5.3. Setelah selesai pada *case folding* dan *formalize*, data kemudian disimpan dalam suatu file *.txt* untuk kemudian digunakan pada tahapan selanjutnya.

```

def formalize(sentence):
    sentence = sentence.replace('\n', " ")
    sentence = sentence.lower()
    sentence = re.sub(r'"'\b'\b"', '', sentence)
    sentence = re.sub(r'([^\A-Za-z])', ' ', sentence)
    tokens = tokenizer.tokenize(sentence)
    word_replace = [word_dict for word_dict in tokens if
word_dict in formalizer_list]
    word_replace = list(set(word_replace))
    word_formalize = [word_formalize for word_formalize in
formalizer_dict if word_formalize[0] in word_replace]
    new_tokens = []
    first = 0
    for word_sentence in sentence.split():
        new_word = word_sentence
        for last_word, formalize in word_formalize:
            if(word_sentence == last_word):
                new_word = formalize
                break
        new_tokens.append(new_word)
    tokens = new_tokens
    for words in tokens:
        if words not in katabaku_dict:
            try:
                is_remove = 0
                for bahasa in detect_langs(words):
                    if 'en' in str(bahasa):
                        is_remove = 1
                if(is_remove == 1):
                    tokens.remove(words)
            except:
                pass
    return tokens

```

Kode 5.3. Kode Program Case Folding, Tokenizing, Dan Formalize

5.3.2 Stemming

Tahapan *stemming* dilakukan untuk mengubah data menjadi kata dasar. Dalam penelitian ini, *stemming* dilakukan dengan menggunakan *library Sastrawi*. Dari hasil *stemming*, kemudian dilakukan pengecekan dengan kamus anti-stem untuk mengubah kembali kata-kata yang mengalami kesalahan pengubahan kata dasar. Kode program yang digunakan *stemming* sesuai dengan Kode 5.4. Setelah proses *stemming* selesai, data kemudian disimpan ke dalam suatu file .txt baru untuk kemudian digunakan pada tahapan selanjutnya.

```

import nltk
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
from nltk.tokenize import RegexpTokenizer

def initiation_dictionary():
    with open('lib-dict/anti_stem.csv', encoding="utf8") as dictionary2:
        for line2 in dictionary2:
            anti_stem.append(line2.replace("\n",
            "").split(','))

def steammer(sentence):
    sentence = stemmer.stem(sentence)

    new_tokens = []
    for word in sentence.split():
        new_word = word
        for anti in anti_stem:
            if(word == anti[0]):
                new_word = anti[1]
                break
        new_tokens.append(new_word)
    tokens = new_tokens

    return tokens

```

Kode 5. 4. Kode Program Stemming dengan *Library Sastrawi*

5.3.3 Stopword Removal

Tahapan *stopword removal* dilakukan dengan menghilangkan kata-kata dianggap tidak memiliki makna serta telah terdaftar pada daftar kata *stopword*. Inisiasi daftar kata *stopword* dilakukan dengan menggunakan kamus *stopword* yang disusun oleh Dr. Eng. Ayu Purwarianti, ST.,MT., et al [15]. Kode yang digunakan untuk melakukan *stopword removal* ditampilkan pada kode 5.5. Setiap data yang telah selesai tahapan *stopword removal* diolah kembali untuk mengetahui kemungkinan terdapat dua atau lebih kata yang memiliki satu makna, seperti lalu lintas, orang tua, dan kata lainnya untuk kemudian digabungkan dengan menggunakan tanda “_”. Kode yang digunakan untuk menghubungkan kata sambung ditampilkan pada kode 5.6. Setelah proses penghubungan kata sambung selesai, data kemudian disimpan ke dalam suatu file .txt baru untuk kemudian digunakan pada tahapan selanjutnya

```
#----- Inisiasi variable -----#
list_stopwords = []

def initiation_dictionary():
    with open('lib-dict/stopword.csv', encoding="utf8") as dictionary3:
        for line3 in dictionary3:
            list_stopwords.append(line3.strip())

def stopword(tokens):
    tokens = [word for word in tokens if not word in list_stopwords]
    return tokens
```

Kode 5. 5. Kode Program *Stopword Removal*

```
data_dict = []
with open('lib-dict/dict_katasambung.txt', encoding="utf8") as myFile:
    for word in myFile:
        data_dict.append(word.replace('\n','').split(';'))
    myFile.close()

with open('hasil/3_post_stopword.txt', encoding="utf8") as myFile:
    docs = myFile.read().splitlines()
    myFile.close()

counter = 1
with open('hasil/4_post_fix.txt', 'w', encoding="utf8") as outfile:
    for line in docs:
        if counter % 100 == 0:
            print('Correcting word in data : ' + str(counter))
        for word_awal,word_koreksi in data_dict:
            if word_awal in line:
                line = line.replace(word_awal,word_koreksi)
        if(line != ''):
            outfile.write(line)
            outfile.write('\n')
        counter += 1
```

Kode 5. 6. Kode Program untuk Menghubungkan Kata Sambung

5.4 Pemodelan Topik dengan Metode *Gaussian LDA*

Pada tahap ini dilakukan pemodelan topik dengan metode *Gaussian Latent Dirichlet Allocation*. Proses pemodelan dilakukan dengan memanfaatkan model *word2vec* yang telah dipilih.

5.4.1 Pemrosesan *Corpus*

Pada tahap ini dilakukan pembuatan *corpus* sebagai data masukan pemodelan topik yang digunakan dalam pada tahap selanjutnya. Setelah *corpus* terbentuk, dilakukan penghapusan *unique tokens* yang muncul terlalu sering dan terlalu jarang muncul dengan menggunakan *dictionary*. Batas *unique tokens* ditentukan berdasarkan hasil perhitungan IDF. Pembuatan corpus dan dictionary ditampilkan dalam Kode 5.7.

```
from gensim.corpora import Dictionary

fileinput = 'hasil/4_post_fix.txt'

curhatan = []
with open(fileinput, errors='ignore', encoding='utf-8') as fid:
    dictt = fid.read().splitlines()
    fid.close()
    for teks in dictt:
        tokens = teks.split(' ')
        curhatan.append(tokens)

dictionary = Dictionary(curhatan)
dictionary.filter_extremes(no_below=31, no_above=0.3)

allpost = []
for post in curhatan:
    sentence = ''
    isfirst = 1
    for word in post:
        if dictionary.doc2bow([word]):
            if(isfirst == 1):
                sentence += word
                isfirst = 0
            else:
                sentence += ' ' + word
    allpost.append(sentence)
```

Kode 5. 7. Pembuatan *Corpus* dan Penghapusan Unique Tokens

Selanjutnya dilakukan inisiasi *variable* untuk menyimpan *corpus*, *dictionary*, serta *variable* lainnya yang akan digunakan untuk pemodelan topik. Inisiasi *variable* ditunjukkan pada Kode 5.8.


```

def __init__(self, num_topics, corpus,
word_vector_filepath=None, word_vector_model=None,
alpha=0.2, outputfile=None, preprocess=False,
run_name=str(1)):
    self.doc_topic_CT = None
    self.corpus = corpus
    self.priors = None
    self.word_vecs = {}
    self.numtopics = num_topics
    self.vocab = set([])
    self.topic_params = defaultdict(dict)
    self.wordvecFP = word_vector_filepath
    self.word_vec_size = None
    self.alpha = alpha
    self.wvmodel = word_vector_model
    self.test_word_topics = defaultdict(list)
    self.test_word_topic_count = defaultdict(int)
    self.word_topics = {}
    self.output_file_name = outputfile
    self.preprocess = preprocess
    self.run_name = run_name

```

Kode 5. 8. Inisiasi Variable

Selanjutnya data diproses menjadi *corpus* dan *dictionary*. Proses ini dilakukan dengan menyimpan data ke dalam *variable docs*. Kemudian setiap kata dikelompokkan secara acak ke dalam masing-masing topik sesuai jumlah topik yang ditentukan. Kode yang digunakan untuk proses menyimpan *vocabulary* ditunjukkan pada Kode 5.9. Hasil pemrosesan *corpus* disimpan di dalam *variable self.corpus* dan *dictionary* disimpan di dalam *variable self.vocab*

```

def process_corpus(self, documents):
    if not self.preprocess:
        temp_corpus = defaultdict(dict)
        random.shuffle(documents) # making sure topics are
        formed semi-randomly
        for index, doc in enumerate(documents):
            words = doc.split()
            temp_corpus[index]['words'] = words
            temp_corpus[index]['topics'] = np.empty(len(words))
    # Random topic assign
    for word in words:
        self.vocab.add(word)
    self.corpus = temp_corpus
    print ("Done processing corpus with {}
documents".format(len(documents)))

    else: # Docs are tokenized and such, just add it into
class
    temp_corpus = defaultdict(dict)
    for idx, doc in enumerate(documents):
        temp_corpus[idx]["words"] = doc
        temp_corpus[idx]["topics"] = np.empty(len(doc))
        for word in words:
            self.vocab.add(word)
        self.corpus = temp_corpus

```

Kode 5. 9. Proses Menyimpan Vocabulary

5.4.2 Pemrosesan Vektor Kata

Pada tahap ini, dilakukan pengubahan kata-kata *corpus* menjadi vektor kata dengan menggunakan daftar vektor kata yang telah dimiliki. Dalam proses ini *file* vektor kata akan disimpan ke dalam *variable self.wordvecFP*, kemudian diambil nilai *word vector* dari model dengan *library gensim.models* menggunakan fungsi *KeyedVectors* dan disimpan pada *variable self.wvmodel*. Selanjutnya, *variable self.word_vec_size* diambil dari *variable wvmodel.vector_size*.

Konversi kata-kata dalam *corpus* dilakukan dengan mengambil kata-kata yang telah tersimpan pada *variable self.vocab* menjadi vektor yang didapatkan pada *self.wvmodel*. Proses konversi kata menjadi vektor ditunjukkan pada Kode 5.10.

```

from gensim.models import KeyedVectors

def process_wordvectors(self, filepath=None):
    if filepath:
        print ("Processing word-vectors, this takes a moment")
        self.wvmodel =
KeyedVectors.load_word2vec_format(fname=filepath, limit=200000)
        useable_vocab = 0
        unusable_vocab = 0
        self.word_vec_size = self.wvmodel.vector_size

        for word in self.vocab:
            try:
                self.word_vecs[word] = self.wvmodel[word] #match
between word on corpus and word embedding
                useable_vocab += 1
            except KeyError:
                self.word_vecs[word] = self.wvmodel['unk']
                unusable_vocab += 1

        print ("There are {0} words that could be converted to word
vectors in your corpus \n" \
"There are {1} words that could NOT be converted to word
vectors".format(useable_vocab, unusable_vocab))
        hasil_training.write("There are " + str(useable_vocab) + "
words that could be converted to word vectors in your corpus
\n" \
"There are " + str(unusable_vocab) + " words that could NOT
be converted to word vectors\n")
    else:
        useable_vocab = 0
        unusable_vocab = 0
        self.word_vec_size = self.wvmodel.vector_size

        for word in self.vocab:
            try:
                self.word_vecs[word] = self.wvmodel[word]
                useable_vocab += 1
            except KeyError:
                self.word_vecs[word] = self.wvmodel['unk']
                unusable_vocab += 1

        print ("There are {0} words that could be converted to word
vectors in your corpus \n" \
"There are {1} words that could NOT be converted to word
vectors".format(useable_vocab, unusable_vocab))
        hasil_training.write("There are " + str(useable_vocab) + "
words that could be converted to word vectors in your corpus
\n" \
"There are " + str(unusable_vocab) + " words that could NOT
be converted to word vectors\n")

```

Kode 5. 10. Proses Konversi Kata Menjadi Vektor.

5.4.3 Pembentukan Model Topik

Pembentukan model dilakukan dengan menginisiasi parameter yang dibutuhkan. Parameter tersebut adalah *prior mean*, *prior co-variance*, *sample means*, *sample co-variances*, dan *document-topic counts*. *Prior means* dan *prior co-variance* merupakan parameter untuk menyimpan *mean* dan *co-variance* pemodelan sebelum dilakukan *Gibbs sampling*. Sedangkan *sample means* dan *sample co-variances* merupakan parameter untuk menyimpan nilai *mean* dan *co-variance* pada saat dilakukan *sampling*. Inisiasi dan perhitungan parameter *prior mean*, *prior co-variance*, *sample means*, *sample co-variances*, dan *document-topic counts* dilakukan dalam fungsi *init()*.

Selanjutnya dilakukan *sampling* dengan metode *Gibbs sampling* untuk mengelompokkan kata-kata yang ada di dalam *corpus* ke dalam topik-topik baru. Langkah pertama yang dilakukan dalam *sampling* adalah melakukan inisiasi *variable* yang dibutuhkan untuk menyimpan kata dan topik dari kata tersebut. Inisiasi *variable* ditampilkan pada Kode 5.11.

```
for docID in self.corpus.keys():
    for idx in range(len(self.corpus[docID]['words'])):
        word = self.corpus[docID]['words'][idx]
        current_topic = self.corpus[docID]['topics'][idx]
```

Kode 5. 11. Inisiasi Variable

Kemudian dilakukan perhitungan ulang dari nilai parameter *Gaussian LDA*. Hal ini dilakukan untuk memperbarui nilai *mean* dan *covariance* dari setiap topik sesuai kata baru yang masuk ke dalam *sampling*. Perhitungan ulang dilakukan dengan 2 tahap, yaitu mengurangi kata dari topik dan menambahkan kata ke dalam topik. Langkah pertama dalam perhitungan ulang adalah memperbarui jumlah kata dalam topik dengan menggunakan fungsi *update_document_topic_counts*. Kode memperbarui jumlah kata dalam topik ditunjukkan pada Kode 5.12.

```

UPDATE_COUNT = True
if UPDATE_COUNT:
    self.update_document_topic_counts(word, topic, docID,
    operation)

def update_document_topic_counts(self, word, topic, docID,
operation):
    if operation == "-":
        self.topic_params[topic]["Topic Sum"] -=
self.word_vecs[word] #jumlah vektor kata dalam topik
minus(-) current_kata
        self.doc_topic_CT[int(docID), int(topic)]-= 1.
#jumlah kata dalam topik per dokumen dikurangi

    if operation == "+":
        self.topic_params[topic]["Topic Sum"] +=
self.word_vecs[word]
        self.doc_topic_CT[docID, topic] += 1.

```

Kode 5. 12. Memperbarui Jumlah Kata dalam Topik

Selanjutnya dilakukan perhitungan ulang terhadap nilai *topic mean* dan *topic co-variance*. Untuk operasi pengurangan, perhitungan *topic co-variance* didapat dari hasil kurang dari pengurangan vektor kata dengan *topic mean* dan dikalikan *scale for recursive downdate*. Kode perhitungan ulang operasi pengurangan ditampilkan pada Kode 5.13.

```

if operation == "-":
    if UPDATE_DISTs:
        wv = self.word_vecs[word]
        mu = self.topic_params[topic]["Topic Mean"]

        centered = wv - mu
        centered *= np.sqrt((kappa_k + 1.) / kappa_k) # Scale
for recursive downdate
        self.topic_params[topic]["Topic          Cov"]      -=
np.outer(centered, centered)

        sample_mean_K      =          self.topic_sample_mean(topic,
topic_count) # V-Bar_k
        topic_mean = ((self.priors.kappa * self.priors.mu) +
(topic_count * sample_mean_K)) / kappa_k # Mu_k

```

Kode 5. 13. Kode Perhitungan Ulang Operasi Pengurangan

Jika operasi yang dilakukan adalah penambahan, maka perhitungan *topic co-variance* didapat dari hasil tambah dari pengurangan vektor kata dengan *topic mean* dan dikalikan *scale*

for recursive downdate. Kode perhitungan ulang operasi penambahan ditampilkan pada Kode 5.14.

```

else:
    sample_mean_K = self.topic_sample_mean(topic, topic_count)
    # V-Bar_k
    topic_mean = ((self.priors.kappa * self.priors.mu) +
                  (topic_count * sample_mean_K)) / kappa_k # Mu_k

    if UPDATE_DISTS:
        centered = (self.word_vecs[word] - topic_mean)
        centered *= np.sqrt(kappa_k / (kappa_k - 1.)) # Scale
    for recursive update
        self.topic_params[topic]["Topic Cov"] +=
        np.outer(centered, centered)

```

Kode 5. 14. Kode Perhitungan Ulang Operasi Penambahan

Kemudian dilakukan perhitungan *log of the probability density* untuk *Student-T Distribution*. Perhitungan *log of the probability density* dilakukan untuk mengetahui nilai *gibbs sampling*. Perhitungan *log of the probability density* dilakukan dengan menjalankan fungsi *draw_new_wt_assgns*. Kode perhitungan *log of the probability density* ditampilkan pada Kode 5.15.

```

def draw_new_wt_assgns(self, word, topic, new_doc=False,
wvmodel=None):
    cov_det = self.topic_params[topic]["Chol Det"]
    Nk = self.topic_params[topic]["Topic Count"]

    # (V_di - Mu)
    centered = self.word_vecs[word] -
self.topic_params[topic]["Topic Mean"]
    d = self.word_vec_size # dimensionality of word vector
    kappa_k = self.topic_params[topic]["Topic Kappa"]

    scaleT = np.sqrt((kappa_k + 1.) / kappa_k * (self.priors.nu
- d + 1.)) # Covariance = chol / sqrt(scaleT)
    nu = self.priors.nu + Nk - d + 1.

    # Covariance Inverse
    cov_inv = self.topic_params[topic]["Topic Inv"]
    cov_inv = centered.T.dot(cov_inv).dot(centered)
    cov_inv *= scaleT

    cov_det = self.topic_params[topic]["Topic Det"]

    a = gammaln((nu + d) / 2.)
    b = (gammaln(nu / 2.) + (d / 2.) * (np.log(nu) +
np.log(pi)) + (0.5 * cov_det) + ((nu + d) / 2.) * np.log(1.
+ cov_inv/nu))
    # Log Multivariate T - PDF
    return a-b

```

Kode 5. 15. Perhitungan *Log of The Probability Density*

Hasil *log of the probability density* yang telah didapat digunakan sebagai masukan dari *gibbs sampling* untuk semua topik yang telah ditentukan jumlahnya. Selanjutnya hasil dari *gibbs sampling* disimpan dalam variable *log_posterior* yang berbentuk array dengan nilai berupa hasil dari log kata untuk masing-masing topik. Kode *gibbs sampling* ditampilkan pada Kode 5.16.

```

log_posterior = np.zeros(self.numtopics)
for k in range(self.numtopics):
    log_pdf = self.draw_new_wt_assgns(word, k)
    Nkd = self.doc_topic_CT[docID, k] # Count of topic in
doc, Ndk
    log_posterior[k] = np.real(np.log(Nkd + self.alpha) +
log_pdf)

```

Kode 5. 16. Kode *Gibbs Sampling*

Kemudian kata pada *corpus* yang telah melalui *gibbs sampling* diproses untuk mengetahui topik tertinggi dari nilai *log_posterior* dan dimasukkan ke dalam topik tertinggi yang dimiliki. Setelah *gibbs sampling* selesai dilakukan pembaruan kata dalam topik dan perhitungan ulang parameter yang dibutuhkan *Gaussian LDA*.

```
max_log_posterior = np.max(log_posterior)
log_posterior -= max_log_posterior
normalized_post = np.exp(log_posterior -
np.log(np.sum(np.exp(log_posterior))))
if MULTINOMIAL_TOPIC_SELECTION:
    new_topic = np.argmax(np.random.multinomial(1,
pvals=normalized_post))
else:
    new_topic = np.argmax(normalized_post)

if not ASSIGN_NEW_TOPICS:
    new_topic = current_topic
self.corpus[docID]['topics'][idx] = new_topic
self.recalculate_topic_params(word, new_topic, docID, "+")
```

Kode 5. 17. Kode Mengambil *log_posterior* Tertinggi, Pembaruan Kata, dan Perhitungan Ulang Parameter *Gaussian LDA*

5.4.4 Eksperimen Pemodelan Topik

Pada tahapan eksperimen, pemodelan topik dengan metode *Gaussian LDA* dilakukan pembentukan model terbaik dengan melakukan eksperimen terhadap parameter masukan, yaitu jumlah topik serta jumlah batas atas dan batas bawah untuk jumlah minimal dan maksimal kemunculan dengan hasil perhitungan IDF.

a. Penentuan Jumlah Topik

Eksperimen untuk menentukan jumlah topik dilakukan dengan beberapa topik, yaitu 5,10,15,20,25,30,35,40, dan 45. Penggunaan jumlah topik dilakukan untuk mendapatkan nilai *perplexity* rendah, dikarenakan semakin rendah nilai *perplexity* menunjukkan kemampuan probabilitas model yang semakin baik.

Langkah awal untuk melakukan perhitungan nilai *perplexity*, dilakukan perhitungan probabilitas kata terhadap

corpus yang dihasilkan dari perhitungan probabilitas kata terhadap topik dikali dengan probabilitas topik terhadap dokumen. Kode perhitungan probabilitas kata ditampilkan pada Kode 5.18.

```

final_prob_w = dict()
readdoc = 1

with open('word_probability/without_stem/pujangga_' +
str(topic) + 'topic_' + str(iteration) + 'iteration' +
keterangan + '.txt', 'w') as final_prob:
    for doc in docs:
        token_word_topic_probability =
extract_topics_new_doc(doc, wv_model)

        p_z_d = dict()
        total_word_doc = 0

        for word,posterior,num_topic in
token_word_topic_probability:
            numtopic = 0
            for prob in posterior:
                if numtopic not in p_z_d:
                    p_z_d[numtopic] = prob
                else:
                    p_z_d[numtopic] += prob

            numtopic += 1
            total_word_doc += 1

        count_while = 0
        while(sum(p_z_d.values()) != 1.0):
            if(count_while == 10):
                break
            for i in range(0,len(p_z_d)):
                p_z_d[i] = p_z_d[i] / sum(p_z_d.values())
            count_while += 1

        for word,posterior,num_topic in
token_word_topic_probability:
            numtopic = 0
            for prob in posterior:
                final_prob.write("{0},{1},{2}\n".format(word,prob,p
_z_d[numtopic]))
            numtopic += 1
        readdoc += 1

```

Kode 5. 18. Kode Perhitungan Probabilitas Kata terhadap Dokumen

Setelah nilai probabilitas kata didapatkan, dilakukan perhitungan nilai *perplexity*. Kode perhitungan nilai *perplexity* ditampilkan pada Kode 5.19.

```
import numpy as np
f = 'word_probability/25topic_stem/pujangga_25topic_' +
str(i) + 'iteration_dengan_batasan_fix' + '.txt'
with open(f, 'r', encoding="utf8") as fi:
    word_prob = fi.read().splitlines()
    fi.close()

a = 0
b = 770520

z = 0
for line in word_prob:
    nilai = np.log10(float(line.split(',')[1]))
    if str(nilai) != 'nan':
        a += np.log10(float(line.split(',')[1]))

c = -(a/b)
d = np.exp(c)
print(d)
```

Kode 5. 19. Kode Perhitungan Nilai *Perplexity*

b. Penentuan Jumlah Iterasi

Setelah jumlah topik terbaik didapatkan, dilakukan penentuan jumlah iterasi terbaik untuk pemodelan topik dengan metode *Gaussian LDA*. Penentuan jumlah iterasi dilakukan dengan membandingkan nilai *perplexity* yang dihasilkan oleh masing-masing model yang dihasilkan dengan iterasi yang berbeda. Perhitungan nilai *perplexity* dilakukan dengan menggunakan Kode 5.18 dan Kode 5.19.

c. Menyimpan Variable Model

Setiap pengujian model yang dilakukan akan disimpan agar model hasil pengujian dapat digunakan kembali tanpa harus melakukan proses pengujian lagi dengan jumlah passes dan topik yang sama. Penyimpanan model hasil pengujian dilakukan dengan menyimpan seluruh *variable* pengujian, yaitu *doc_topic_CT*, *corpus*, *word_vecs*, *numtopics*, *vocab*, *topic_params*, *wordvecFP*, *word_vec_size*, *alpha*, *wvmodel*, *test_word_topics*, *test_word_topics_count*, *word_topics*,

output_file_name, *preprocess*, *run_name*, *priors_nu*, *priors_kappa*, *priors_psi*, dan *priors_mu*. Penyimpanan *variable* pengujian ini dilakukan dengan menggunakan *library pickle*. Kode yang digunakan untuk menyimpan model ditampilkan pada kode 5.20.

```
doc_topic_CT = open(foldername + '/doc_topic_CT' + '_' + str(i)
+ 'iter', 'wb')
pickle.dump(self.doc_topic_CT, doc_topic_CT)

corpus = open(foldername + '/corpus' + '_' + str(i) + 'iter',
'wb')
pickle.dump(self.corpus, corpus)

# priors = open(foldername + '/priors', 'wb')
# pickle.dump(self.priors, priors)

word_vecs = open(foldername + '/word_vecs' + '_' + str(i) +
'iter', 'wb')
pickle.dump(self.word_vecs, word_vecs)

numtopics = open(foldername + '/numtopics' + '_' + str(i) +
'iter', 'wb')
pickle.dump(self.numtopics, numtopics)

vocab = open(foldername + '/vocab' + '_' + str(i) + 'iter', 'wb')
pickle.dump(self.vocab, vocab)

topic_params = open(foldername + '/topic_params' + '_' + str(i)
+ 'iter', 'wb')
pickle.dump(self.topic_params, topic_params)

wordvecFP = open(foldername + '/wordvecFP' + '_' + str(i) +
'iter', 'wb')
pickle.dump(self.wordvecFP, wordvecFP)

word_vec_size = open(foldername + '/word_vec_size' + '_' + str(i)
+ 'iter', 'wb')
pickle.dump(self.word_vec_size, word_vec_size)

alpha = open(foldername + '/alpha' + '_' + str(i) + 'iter', 'wb')
pickle.dump(self.alpha, alpha)

wvmodel = open(foldername + '/wvmodel' + '_' + str(i) + 'iter',
'wb')
pickle.dump(self.wvmodel, wvmodel)
```

```

test_word_topics = open(foldername + '/test_word_topics' + '_' +
str(i) + 'iter', 'wb')
pickle.dump(self.test_word_topics, test_word_topics)

test_word_topic_count = open(foldername +
'/test_word_topic_count' + '_' + str(i) + 'iter', 'wb')
pickle.dump(self.test_word_topic_count, test_word_topic_count)

word_topics = open(foldername + '/word_topics' + '_' + str(i) +
'iter', 'wb')
pickle.dump(self.word_topics, word_topics)

output_file_name = open(foldername + '/output_file_name' + '_' +
str(i) + 'iter', 'wb')
pickle.dump(self.output_file_name, output_file_name)

preprocess = open(foldername + '/preprocess' + '_' + str(i) +
'iter', 'wb')
pickle.dump(self.preprocess, preprocess)

run_name = open(foldername + '/run_name' + '_' + str(i) + 'iter',
'wb')
pickle.dump(self.run_name, run_name)

priors_nu = open(foldername + '/priors_nu' + '_' + str(i) +
'iter', 'wb')
pickle.dump(self.priors.nu, priors_nu)

priors_kappa = open(foldername + '/priors_kappa' + '_' + str(i)
+ 'iter', 'wb')
pickle.dump(self.priors.kappa, priors_kappa)

priors_psi = open(foldername + '/priors_psi' + '_' + str(i) +
'iter', 'wb')
pickle.dump(self.priors.psi, priors_psi)

priors_mu = open(foldername + '/priors_mu' + '_' + str(i) +
'iter', 'wb')
pickle.dump(self.priors.mu, priors_mu)

```

Kode 5. 20. Kode Program Menyimpan Variable Pemodelan *Gaussian LDA*

5.5 Pemodelan Topik dengan Metode *Latent Dirichlet Allocation*

Tahap pemodelan topik dengan menggunakan metode *Latent Dirichlet Allocation* merupakan tahapan membentuk

model dan melakukan eksperimen model untuk mendapatkan model yang paling baik.

5.5.1 Pembentukan *Dictionary*

Pembentukan *dictionary* menggunakan *library gensim.corpora* dengan modul *Dictionary* dilakukan untuk memberikan nilai unik berupa integer terhadap setiap kata pada seluruh data yang digunakan. Kode yang digunakan untuk memproses data dan membentuk *dictionary* ditampilkan pada kode 5.21.

```
from gensim.corpora import Dictionary

def load_dataset(filename):

    with open(fileinput, errors='ignore', encoding='utf-8') as
    fid:
        dictt = fid.read().splitlines()
        fid.close()

    for teks in dictt:
        tokens = teks.split(',')
        curhatan.append(tokens)

fileinput = 'data/4_post_fix.txt'
#----- 1. Load file hasil preprocessing -----
-----#
load_dataset(fileinput)
#----- 2. Membuat term dictionary dari korpus,
setiap kata unik akan diberikan index -----#
dictionary = Dictionary(curhatan)
```

Kode 5. 21. Kode Program untuk Memproses Data dan Membentuk *Dictionary*

Setelah *dictionary* terbentuk, dilakukan penghapusan *unique tokens* yang muncul terlalu sering dan terlalu jarang muncul. Batas *unique tokens* ditentukan berdasarkan hasil perhitungan IDF. Kode yang digunakan untuk menghapus *unique tokens* ditampilkan pada kode 5.22.

```
max_freq = 0.3
min_freq = 31
dictionary.filter_extremes(no_below=min_freq, no_above=max_freq)
```

Kode 5. 22. Kode Program untuk Mengatur Batasan *Unique Tokens* pada *Dictionary*

5.5.2 Pembentukan Dokumen Matriks atau *Corpus*

Pembentukan *corpus* dilakukan berdasarkan *dictionary* yang sudah dibentuk terhadap token yang ada pada seluruh data. Kode yang digunakan untuk pembentukan *corpus* ditampilkan pada kode 5.23.

```
corpus = [dictionary.doc2bow(curhat) for curhat in curhatan]
```

Kode 5. 23. Kode Program untuk Pembentukan *Corpus*

5.5.3 Pembentukan Model Topik

Tahap awal dari pemodelan topik adalah pembentukkan model topik. Untuk membentuk model topik, digunakan library dari *gensim.models* modul *ldamodel.LdaModel*. Beberapa parameter masukan yang harus diperhatikan dalam pembentukan model topik yaitu jumlah topik dan jumlah *passes*. Dari kedua nilai parameter masukan ini akan nantinya dilakukan eksperimen untuk mendapatkan model yang terbaik. Kode yang digunakan untuk pembentukan model ditampilkan pada kode 5.24.

```
lda = ldamodel.LdaModel(corpus, num_topics=20,  
id2word=dictionary, passes=100)
```

Kode 5. 24. Kode Program Pembentukan Model LDA

5.5.4 Eksperimen Pemodelan Topik dengan *LDA Model*

Pada tahapan eksperimen, pemodelan topik dengan metode *Latent Dirichlet Allocation* dilakukan pembentukan model terbaik dengan melakukan eksperimen terhadap parameter masukan, yaitu *passes* dan jumlah topik. Berdasarkan hal ini, eksperimen akan dibedakan ke dalam dua tahap, yaitu tahap penentuan jumlah *passes* dan penentuan jumlah topik.

a. Penentuan Jumlah *Passes*

Passes pada metode *ldamodel.LdaModel* merupakan perulangan yang dilakukan model untuk melakukan pembelajaran. Penentuan jumlah *passes* dalam menjalankan eksperimen pemodelan topik merupakan komponen penting,

hal ini dilakukan untuk mendapatkan perulangan yang tepat sehingga dapat menghasilkan model dengan pembelajaran dengan baik. Jumlah *passes* yang terlalu sedikit dalam pembelajaran model menyebabkan model belum dapat mencapai konvergen untuk mendistribusikan seluruh kata-kata yang mirip atau satu topik. Sedangkan jumlah *passes* yang terlalu banyak akan menyebabkan model menjadi *overfitting*.

Eksperimen untuk mendapatkan jumlah *passes* yang tepat dilakukan 10 kali dengan jumlah topik yang berbeda-beda. Hal ini dilakukan untuk mendapatkan jumlah *passes* yang stabil. Jumlah topik yang digunakan adalah 2, 4, 6, 8, 10, 12, 14, 16, 18, dan 20 topik. Berdasarkan eksperimen ini, nilai *perplexity* akan muncul ketika pembelajaran dengan kelipatan 20.000 dokumen dan ketika seluruh dokumen selesai setiap *passes* yang dijalankan. Analisis akan dilakukan terhadap nilai *perplexity* ketika seluruh dokumen selesai dijalankan untuk mengetahui tren nilai *perplexity* setiap *passes*.

Untuk mendapatkan nilai *perplexity* di setiap *passes*, dibutuhkan *library logging* sehingga hasil yang keluar tercatat pada *log*. Selanjutnya, nilai *perplexity* setiap *passes* akan divisualisasikan menggunakan *line chart* di *Microsoft Excel*. Berdasarkan visualisasi yang dilakukan, dapat dilihat nilai *perplexity* mulai stabil pada *passes* tertentu. *Passes* yang digunakan adalah *passes* dengan nilai *perplexity* paling awal menunjukkan tren yang stabil dari seluruh jumlah topik yang diuji. Kode yang digunakan untuk eksperimen jumlah *passes* ditampilkan pada kode 5.25.

```

import logging
from gensim.models import LdaModel

jumlahTopik = 20
jumlahPasses = 100

logging.basicConfig(format='%(asctime)s    :   %(levelname)s    :
%(message)s', level=logging.INFO)

lda = LdaModel(LdaModel(corpus, num_topics=jumlahTopik,
id2word=dictionary, passes=jumlahPasses)

```

Kode 5. 25. Kode Program Penentuan Jumlah *Passes*

b. Penentuan Jumlah Topik

Setelah mendapatkan jumlah *passes* yang sesuai, dilanjutkan dengan eksperimen penentuan jumlah topik. Dalam melakukan eksperimen, dilakukan percobaan dengan jumlah topik 2, 3, 4, 5, 6, 7, 8, 9, dan 10. Hal ini dilakukan untuk mendapatkan nilai *perplexity* terendah. Semakin rendah nilai *perplexity* menunjukkan kemampuan probabilitas prediksi model semakin baik, dan sebaliknya. Eksperimen diulangi sebanyak 30 kali, kemudian dihitung rata-rata dan standar deviasi dari 30 kali pengujian tersebut.

Nilai rata-rata dan standar deviasi *perplexity* yang didapat kemudian divisualisasikan menggunakan *bar chart* di dalam *Microsoft Excel*. Nilai rata-rata dan standar deviasi paling rendah menunjukkan model yang lebih baik.

Kode yang digunakan untuk eksperimen jumlah topik ditampilkan pada kode 5.26.

```

import logging
from gensim.models import LdaModel

jumlahTopik = 20
jumlahPasses = 100

logging.basicConfig(format='%(asctime)s    :   %(levelname)s    :
%(message)s', level=logging.INFO)

lda = LdaModel(LdaModel(corpus, num_topics=jumlahTopik,
id2word=dictionary, passes=jumlahPasses)

```

Kode 5. 26. Kode Program Penentuan Jumlah Topik

c. Menyimpan Model

Setiap pengujian model yang dilakukan disimpan kedalam *file* dengan format *.model*. Hal ini dilakukan agar model hasil pengujian dapat digunakan kembali tanpa harus melakukan proses pengujian lagi dengan jumlah *passes* dan topik yang sama. Kode yang digunakan untuk menyimpan model ditampilkan pada kode 5.27.

```
jumlahTopik = 6
jumlahPasses = 50
no_uji_model = 5
max_freq = 0.3
min_freq = 31

lda.save('model/lda_' + str(jumlahTopik) + 'topics_' +
str(jumlahPasses) + 'passes_' + str(min_freq) + 'min_freq_' +
str(no_uji_model) + '.model')
```

Kode 5. 27. Kode Program Penentuan Jumlah Topik

5.6 Validasi Model Topik *Gaussian LDA*

Setelah analisis nilai *perplexity* dilakukan dan didapatkan model yang dianggap baik berdasarkan nilai *perplexity*, selanjutnya dilakukan validasi model topik dengan menggunakan nilai *topic coherence*. Validasi *topic coherence* dilakukan dengan menggunakan *pointwise mutual information*. Validasi ini juga dilakukan terhadap model *Latent Dirichlet Allocation* untuk membandingkan nilai *topic coherence* model kedua jenis metode dengan *PMI*.

Untuk melakukan validasi dengan menggunakan metode *pointwise mutual information*, diambil 5 kata dengan probabilitas tertinggi pada masing-masing topik dari model terbaik yang dihasilkan. Pemilihan kata dilakukan dengan menggunakan *Microsoft Excel* dari hasil kata dan probabilitas kata yang telah disimpan pada tahap sebelumnya.

Perhitungan *pointwise mutual information* didapatkan dengan menghitung probabilitas kemunculan kedua kata secara bersamaan dibanding jumlah perkalian probabilitas kemunculan kata pertama dan probabilitas kemunculan kata

kedua. Kode perhitungan *pointwise mutual information* ditampilkan pada Kode 5.28.

```
word1 = "peduli"
word2 = "keluarga"
docs = [w.replace(',', ' ') for w in docs]
for doc in docs:
    find_word1 = 0
    find_word2 = 0

    token = doc.split(' ')

    if word1 in token:
        find_word1 = 1
        tf_word1 += 1
    if word2 in token:
        find_word2 = 1
        tf_word2 += 1
    if (find_word1 == 1 and find_word2 == 1):
        total_coocurrence += 1

    total_doc += 1
    total_word += len(token)

px = tf_word1/total_word
py = tf_word2/total_word
pxy = total_coocurrence/total_doc
pmi = np.log10(pxy / (px * py))
```

Kode 5. 28. Perhitungan Pointwise Mutual Information

Setelah seluruh hasil *pointwise mutual information* didapat, dihitung rata-rata nilai *pointwise mutual information* dari masing masing topik untuk mengetahui nilai *pointwise mutual information* dari masing masing topik. Nilai *pointwise mutual information* model selanjutnya dihitung dari rata-rata *pointwise mutual information* masing-masing topik.

5.7 Validasi Model Topik *Latent Dirichlet Allocation*

Setelah mendapatkan model terbaik sesuai dengan nilai *perplexity* yang didapat, dilakukan validasi model topik dengan menggunakan nilai *topic coherence score* setiap model. Sebelum menghitung *topic coherence score*, dilakukan *load* model dari hasil pengujian dan diambil kata dalam masing-masing topik. Kode yang digunakan untuk *load* model dan pengambilan kata dalam topik ditampilkan pada kode 5.29.

```

import gensim
from gensim.corpora import Dictionary, MmCorpus
from gensim.models import CoherenceModel, LdaModel

file_dir = "model/lda_10topics_50passes_13.model"
curhatan = []
corpus = []

def load_dataset(filename):
    file = open(filename, 'r')
    lines = file.read().split('\n')

    for line in lines:
        words = line.split(',')
        datasentence = []
        for word in words:
            datasentence.append(word)
            curhatan.append(datasentence)

class coherence_check(object):
    load_dataset('data/4_post_fix.txt')
    dictionary = Dictionary(curhatan)
    dictionary.filter_extremes(no_below=31, no_above=0.3)
    for curhat in curhatan:
        corpus.append(dictionary.doc2bow(curhat))
    lda = LdaModel.LdaModel.load(file_dir, mmap='r')
    topics_matrix = lda.show_topics(formatted=False,
num_words=30)
    topics = []
    for topic_no, topic_words in topics_matrix:
        new_topics = []
        for word, prob in topic_words:
            new_topics.append(word)
        topics.append(new_topics)

```

Kode 5. 29. Kode Program Load Model dan Pengambilan Kata Dalam Topik

Setelah kata dalam topik didapatkan, kemudian dilakukan perhitungan *topic coherence* score. Perhitungan dilakukan menggunakan *library* `gensim.models` modul `CoherenceModel` Kode yang digunakan untuk menghitung *topic coherence score* ditampilkan pada kode 5.30.

```

cm = CoherenceModel(topics=topics, corpus=corpus,
dictionary=dictionary, coherence='u_mass')
cm.get_coherence()

```

Kode 5. 30. Kode Program Penentuan Jumlah Topik

5.8 Konstruksi Perangkat Lunak

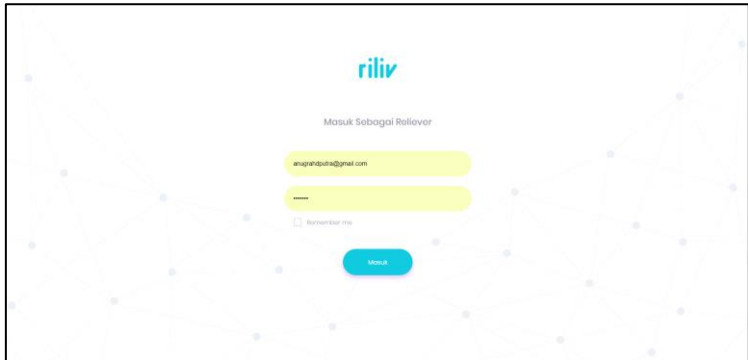
Implementasi aplikasi dengan nama portal reliever dalam penelitian ini merupakan dashboard yang dihasilkan dari *posting* atau data konseling. Dashboard dibuat untuk menunjukkan probabilitas topik terhadap setiap *posting*. Visualisasi probabilitas topik ditampilkan dalam bentuk grafik *pie chart* dengan menggunakan *library amCharts*.

Pada Gambar 5.1 menjelaskan antarmuka aplikasi berupa tampilan halaman masuk yang digunakan pengguna untuk masuk ke dalam aplikasi.

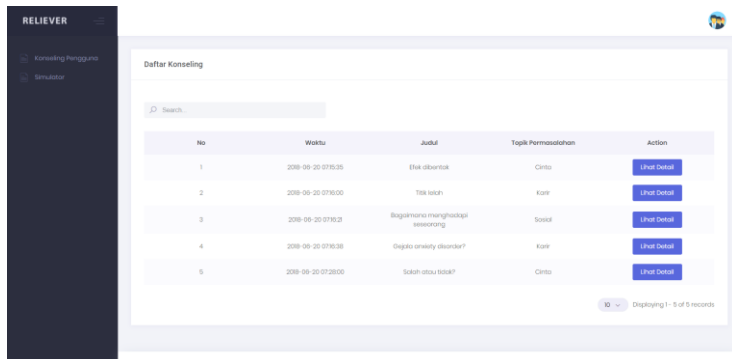
Pada Gambar 5.2 menjelaskan antarmuka aplikasi berupa daftar konseling yang telah masuk ke dalam aplikasi. Daftar konseling ditampilkan ke dalam tabel dengan tombol “Lihat Detail” untuk memindahkan pengguna ke halaman detail konseling yang dipilih.

Pada Gambar 5.3 menjelaskan antarmuka aplikasi berupa detail konseling yang telah dipilih dari halaman sebelumnya. Pada halaman detail konseling ditampilkan visualisasi probabilitas topik terhadap teks konseling dengan menggunakan *pie chart*. Bagian Saran Topik digunakan pengguna untuk memberikan saran pembenaran topik yang sesuai dengan teks konseling. Kemudian bagian Obrolan menampilkan judul, waktu, serta teks dari konseling yang dipilih.

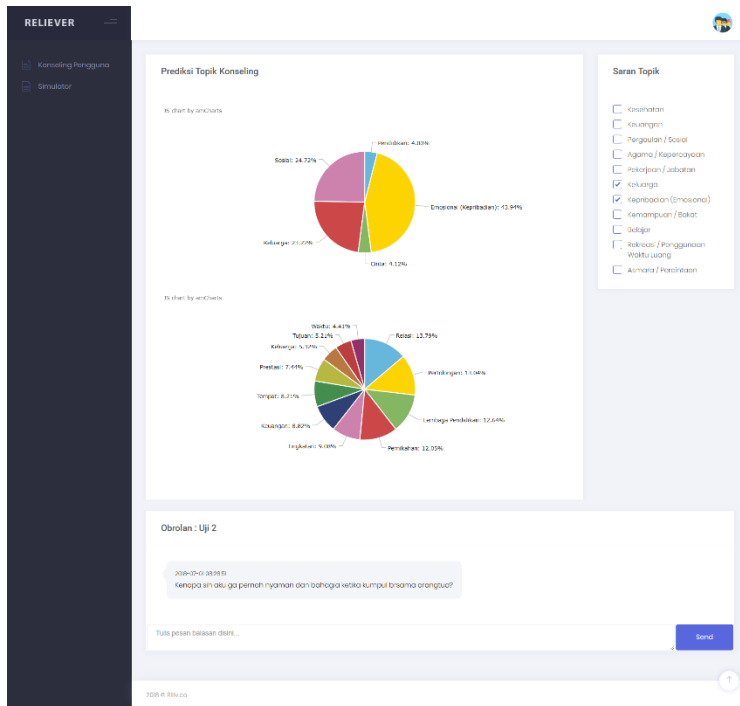
Pada Gambar 5.4 menjelaskan antarmuka aplikasi berupa halaman simulasi. Halaman ini digunakan untuk memasukkan data konseling baru yang kemudian akan dilakukan perhitungan probabilitas topik terhadap konseling tersebut.



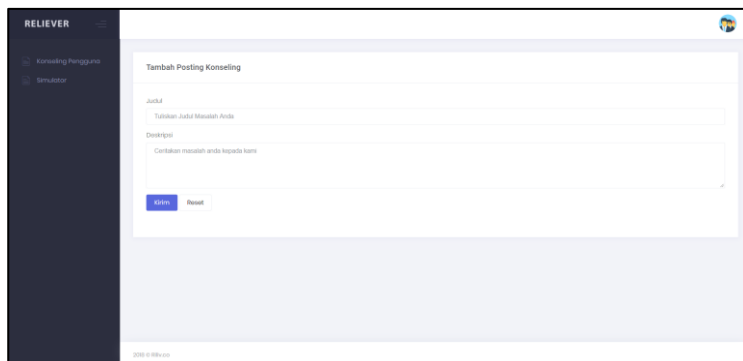
Gambar 5. 1 Antar Muka Aplikasi Halaman Masuk



Gambar 5. 2. Antar Muka Aplikasi Halaman Daftar Konseling



Gambar 5. 3. Antar Muka Aplikasi Halaman Detail Konseling



Gambar 5. 4. Antar Muka Aplikasi Halaman *Simulator Posting*

BAB VI

HASIL DAN PEMBAHASAN

Pada bab ini akan dijelaskan proses pengujian dan analisis terhadap hasil pengujian yang diperoleh dari proses implementasi yang telah dibahas pada bab sebelumnya.

6.1 Mempersiapkan Data

Pemuatan data yang dilakukan pada penelitian ini memiliki jumlah keseluruhan 113975 data. Untuk gambaran jumlah data dapat dilihat pada Tabel 6.1.

Tabel 6. 1. Jumlah Data Konseling

Bulan	Jumlah
Januari	11204
Februari	5112
Maret	8293
April	11933
Mei	14549
Juni	11317
Juli	9992
Agustus	10074
September	9873
Oktober	7447
November	6546
Desember	7635
Total	113975

Data pada bulan Januari 2017 berjumlah 11204 data. Data pada bulan Februari 2017 berjumlah 5112 data. Data pada bulan Maret 2017 berjumlah 8293 data. Data pada bulan April 2017 berjumlah 119933 data. Data pada bulan Mei 2017 berjumlah 14549 data. Data pada bulan Juni 2017 berjumlah

11317 data. Data pada bulan Juli 2017 berjumlah 9992 data. Data pada bulan Agustus 2017 berjumlah 10074 data. Data pada bulan September 2017 berjumlah 9873 data. Data pada bulan Oktober 2017 berjumlah 6546 data. Data pada bulan November 2017 berjumlah 132 data. Data pada bulan Desember 2017 berjumlah 7635 data.

6.2 Pra-Proses Data

Tahap ini dilakukan dengan menghapus data yang memiliki duplikat. Kemudian dilakukan *case folding*, *tokenizing*, *formalize*, *stemming*, dan *stopword removal* dengan dua skenario, yaitu dengan *stemming* dan tanpa *stemming*. Perubahan jumlah kata setelah dilakukan pra-proses dapat dilihat pada tabel 6.2.

Tabel 6. 2. Jumlah Kata dalam Data Konseling

Tahapan	Stemming	Tanpa Stemming
Jumlah Asli	7547071	7547071
Penghapusan Data Duplikat	2671634	2671634
<i>Formalizing, Stemming</i>	2694752	-
<i>Stopword Removal</i>	923817	955512
Pemisah Kata Hubung	900156	937491

6.3 Pemodelan Topik dengan *Gaussian LDA*

Pemodelan dengan *Gaussian Latent Dirichlet Allocation* dilakukan dengan menggunakan 2 jenis skenario eksperimen. Untuk menghasilkan model yang terbaik dari dua jenis skenario, dilakukan pembentukan *corpus*, vektor kata, dan penentuan jumlah topik model.

6.3.1 Pembentukan *Corpus*

Pada tahap ini dilakukan pembentukan *corpus*, kemudian data corpus dikelompokkan secara acak sesuai jumlah topik yang ditentukan. Selanjutnya dilakukan penghapusan *unique tokens* sesuai batasan yang ditentukan. Jumlah *unique tokens* yang dihasilkan dapat dilihat pada tabel 6.3.

Tabel 6. 3. Jumlah *Unique Tokens*

Tahapan	Stemming	Tanpa Stemming
Seluruh <i>unique tokens</i>	42615	56001
Min = 31 Max = 0.3	1800	2908

6.3.2 Pembentukan Vektor Kata

Pada tahap ini dilakukan konversi kata ke dalam bentuk vektor. Kata-kata yang terdapat pada model akan dihitung sebagai kata *useable_vocab*, sedangkan kata-kata yang tidak terdapat pada model akan dihitung sebagai kata *unusable_vocab* dan nilai vektornya dikonversikan menjadi vektor kata '*unk*' yaitu nilai vektor yang disediakan untuk kata-kata diluar model (*unknown*). Data jumlah kata *usable_vocab* dan *unusable_vocab* ditampilkan pada Tabel 6.4.

Tabel 6. 4. Jumlah Kata *usable_vocab* dan *unusable_vocab*.

Tahapan	Stemming	Tanpa Stemming
<i>usable_vocab</i>	1574	2682
<i>unusable_vocab</i>	191	169

6.3.3 Penentuan Jumlah Topik

Penentuan jumlah topik dalam pemodelan topik *Gaussian LDA* dilakukan dengan menganalisis nilai *perplexity*. Dalam penentuan topik dilakukan eksperimen dengan menggunakan beberapa topik, yaitu 5, 10, 15, 20, 25, 30, 35, 40, dan 45 topik. Eksperimen untuk menentukan jumlah topik dilakukan untuk dua jenis skenario yang telah ditentukan.

a. *Stemming*

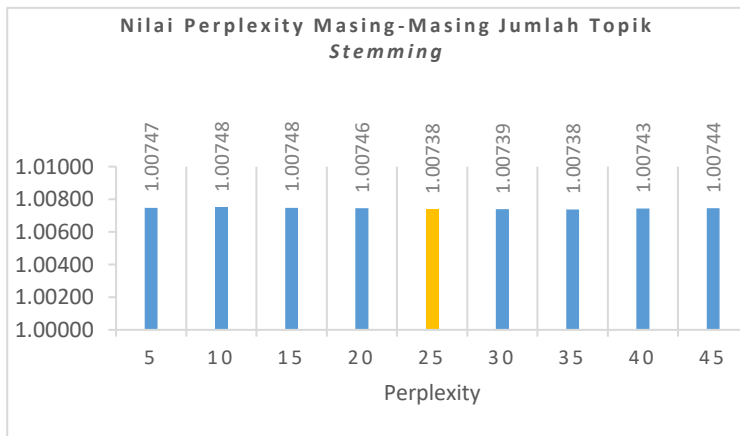


Diagram 6. 1. Nilai *Perplexity* Jumlah Topik Uji *Stemming*

Diagram 6.1 memperlihatkan nilai *perplexity* yang paling rendah pada skenario *stemming* berada di 25 topik dengan nilai 1.00738. Apabila dibandingkan dengan nilai *perplexity* seluruh jumlah topik uji yang lain, perbedaan nilai *perplexity* sangat dekat. Berdasarkan hasil diatas disimpulkan bahwa model dengan 25 topik merupakan model terbaik dalam skenario data dengan *stemming*.

b. Tanpa Stemming

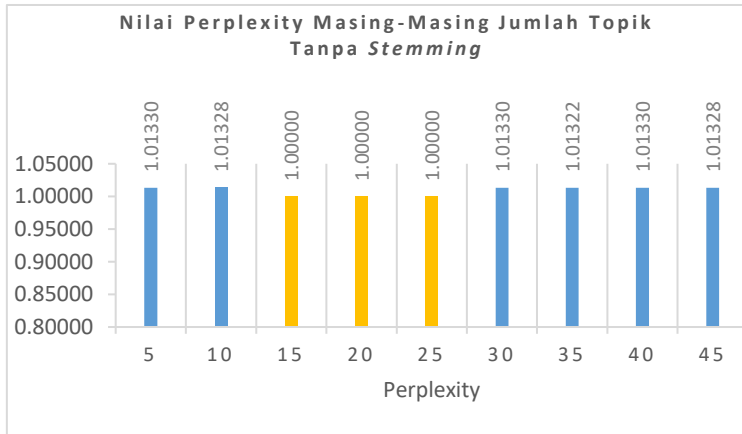


Diagram 6. 2. Nilai Perplexity Jumlah Topik Uji Tanpa Stemming

Diagram 6.2 memperlihatkan nilai *perplexity* yang paling rendah pada skenario tanpa *stemming* berada di 15, 20, dan 25 topik dengan nilai 1.0. Apabila dibandingkan dengan nilai *perplexity* seluruh jumlah topik uji yang lain, perbedaan nilai *perplexity* sangat sedikit. Karena hasil yang didapat antara 15, 20, dan 25 topik sama, maka dipilih 25 topik sehingga jumlah topik antara skenario dengan *stemming* dan tanpa *stemming* sama.

6.3.4 Penentuan Jumlah Iterasi

Penentuan jumlah *passes* dilakukan berdasarkan analisis terhadap kestabilan nilai *perplexity* setiap iterasi. Percobaan yang dilakukan dengan menggunakan nilai iterasi awal 30. Jumlah topik yang digunakan adalah 25 topik dengan *stemming*. Hasil nilai *perplexity* yang muncul dari hasil eksperimen kemudian dicatat dan ditampilkan dalam Diagram 6.3.

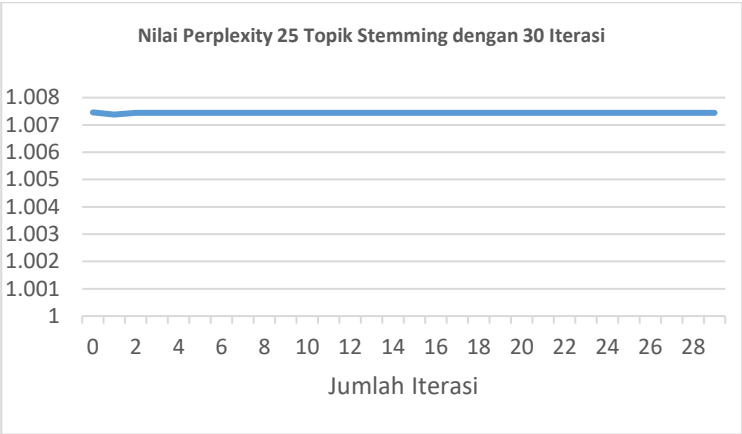


Diagram 6. 3. Nilai Perplexity 25 Topik Stemming dengan 30 iterasi.

Dalam 30 iterasi yang ditentukan pada eksperimen, didapatkan pergerakan nilai *perplexity* yang stabil sejak iterasi pertama. Dari hasil diatas, diputuskan bahwa nilai *iterasi* yang digunakan pada tahap selanjutnya adalah 2 iterasi.

6.4 Pemodelan Topik dengan Metode *Latent Dirichlet Allocation*

Pada tahap pemodelan topik dengan metode *Latent Dirichlet Allocation*, terdapat 2 skenario yang akan dilakukan dalam eksperimen berdasarkan 2 jenis data yang telah disiapkan.

6.4.1 Pembentukan *Dictionary*

Pada tahap ini, seluruh token diambil secara unik sehingga menghasilkan *unique tokens* yang mewakili kata pada keseluruhan token. Jumlah *unique tokens* dapat dilihat pada tabel 6.4.

Tabel Diagram 6. 4. Jumlah Data Konseling

Tahapan	Stemming	Tanpa Stemming
Seluruh <i>unique tokens</i>	43964	57965

Min = 31	1508	2904
Max = 0.3		

Jumlah akhir *unique tokens* ini kemudian digunakan sebagai *dictionary* pada proses pemodelan di tahap selanjutnya.

6.4.2 Penentuan Jumlah *Passes*

Penentuan jumlah *passes* dilakukan berdasarkan analisis terhadap kestabilan nilai *perplexity* setiap *passes*. Percobaan yang dilakukan dengan menggunakan nilai *passes* awal 100. Jumlah topik yang digunakan adalah 2, 4, 6, 8, 10, 12, 14, 16, 18, dan 20 topik. Hasil nilai *perplexity* yang muncul dari hasil eksperimen kemudian dicatat dan ditampilkan dalam Diagram 6.5.

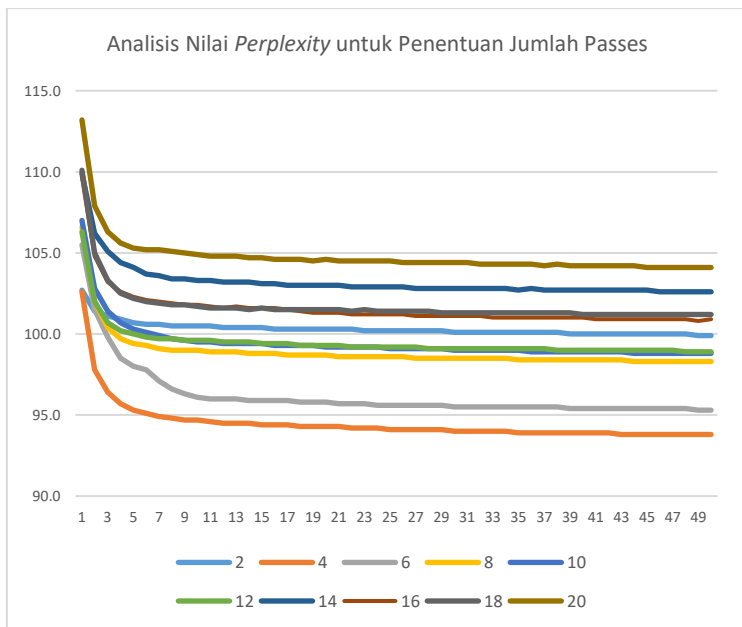


Diagram 6. 5. Analisis Nilai *Perplexity* untuk Penentuan Jumlah *Passes*

Dalam 100 *passes* yang ditentukan pada eksperimen, didapatkan 50 *passes* telah memiliki nilai perplexity yang stabil. Pada diagram di atas dapat dilihat bahwa pergerakan nilai

perplexity telah mulai stabil mulai *passes* ke-35. Dari hasil diatas, diputuskan bahwa nilai *passes* yang digunakan pada tahap selanjutnya adalah 50 *passes*. Hasil eksperimen penentuan jumlah *passes* ditampilkan pada Lampiran A.

6.4.3 Penentuan Jumlah Topik

Pada penentuan jumlah topik, juga dilakukan analisis terhadap nilai *perplexity*. Nilai *perplexity* yang diperhatikan untuk penentuan jumlah *passes* adalah nilai *perplexity* setiap *passes* untuk dilihat kestabilannya, sedangkan untuk penentuan jumlah topik yang diperhatikan adalah nilai *perplexity* terakhir yaitu pada *passes* ke-50. Selain itu, dalam penentuan jumlah topik, eksperimen dilakukan dalam rentang jumlah topik lebih kecil, yaitu dimulai dengan 2 topik hingga 10 topik dengan selisih 1 topik pada masing-masing percobaannya. Eksperimen penentuan jumlah topik dilakukan pada kedua skenario, yaitu skenario tanpa *stemming* dan dengan *stemming*. Hasil eksperimen nilai *perplexity* untuk menentukan jumlah topik pada skenario dengan *stemming* dan tanpa *stemming* dapat dilihat pada Lampiran A.

a. *Stemming*

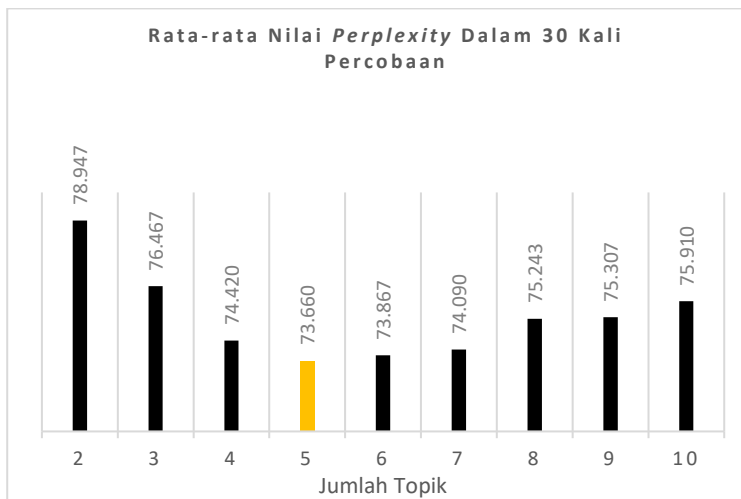


Diagram 6. 6. Rata-rata nilai *perplexity* dalam 30 kali percobaan

Pada skenario data dengan *stemming*, dapat dilihat pada Diagram 6.6 bahwa model dengan rata-rata nilai *perplexity* paling rendah adalah model dengan jumlah topik 5, yaitu sebesar 73.66. Pada jumlah topik 2, rata-rata nilai *perplexity* lebih tinggi dibandingkan pada jumlah topik yang lainnya dan terus menurun hingga jumlah topik 5. Selanjutnya diketahui rata-rata nilai *perplexity* kembali meningkat hingga jumlah topik 10 walaupun dengan tidak signifikan, sehingga model dengan 5 topik merupakan model terbaik dalam skenario data dengan *stemming*.

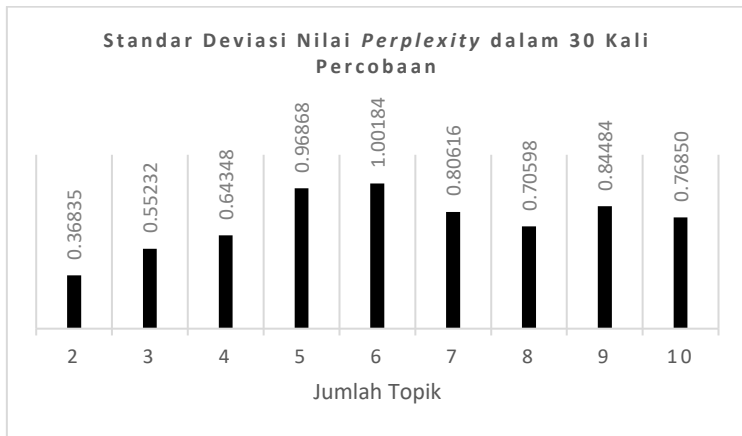


Diagram 6. 7. Standar Deviasi nilai *perplexity* dalam 30 kali percobaan

Selanjutnya berdasarkan Diagram 6.7, standar deviasi nilai *perplexity* model dengan jumlah topik 5 adalah 0.96868. Jika dibandingkan dengan rata-rata nilai standar deviasi keseluruhan, yaitu 0.74002, nilai standar deviasi topik 5 lebih tinggi dari rata-rata nilai standar deviasi keseluruhan.

b. Tanpa Stemming

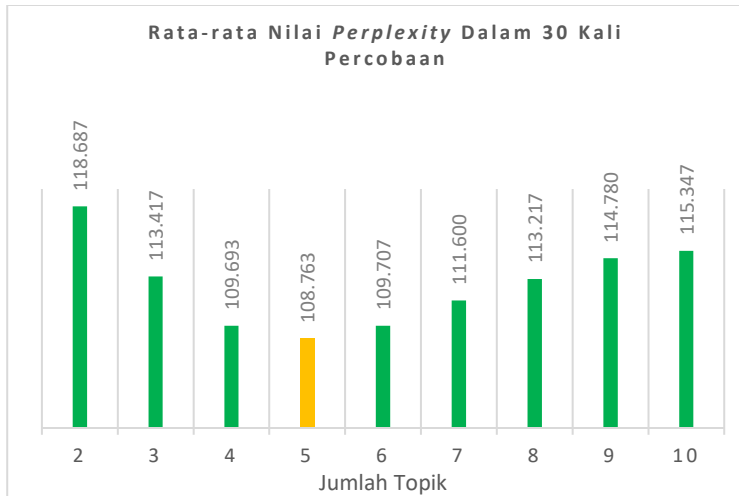


Diagram 6. 8. Rata-rata nilai *perplexity* dalam 30 kali percobaan

Pada skenario data dengan tanpa *stemming*, dapat dilihat pada Diagram 6.8 bahwa model dengan rata-rata nilai *perplexity* paling rendah adalah model dengan jumlah topik 5, yaitu sebesar 108.763. Pada jumlah topik 2, rata-rata nilai *perplexity* lebih tinggi dibandingkan pada jumlah topik yang lainnya dan terus menurun hingga jumlah topik 5. Selanjutnya diketahui rata-rata nilai *perplexity* kembali meningkat hingga jumlah topik 10 walaupun dengan tidak signifikan, sehingga model dengan 5 topik merupakan model terbaik dalam skenario data dengan tanpa *stemming*.

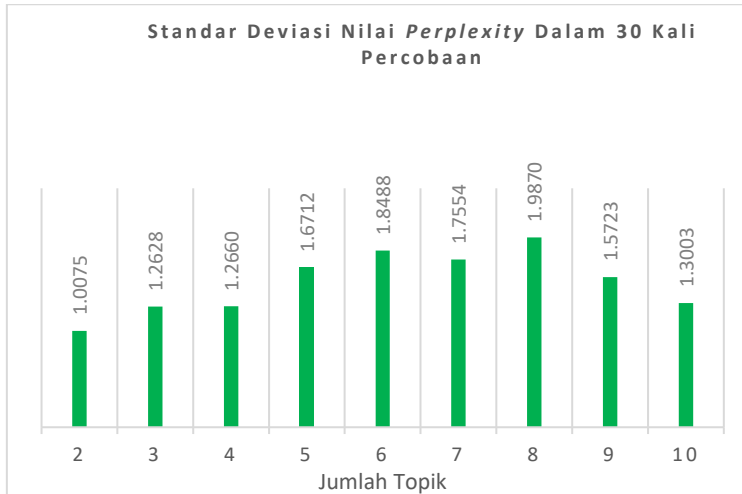


Diagram 6. 9. Standar Deviasi nilai *perplexity* dalam 30 kali percobaan

Selanjutnya berdasarkan Diagram 6.9, standar deviasi nilai *perplexity* model dengan jumlah topik 5 adalah 1.6712. Jika dibandingkan dengan rata-rata nilai standar deviasi keseluruhan, yaitu 1.5190, nilai standar deviasi topik 5 lebih tinggi dari rata-rata nilai standar deviasi keseluruhan.

6.5 Validasi Model Topik *Latent Dirichlet Allocation*

Pada validasi model topik dilakukan perhitungan *topic coherence* dari model yang dihasilkan. Hasil jumlah topik yang didapat pada masing-masing skenario sama, sehingga jumlah topik yang akan dibandingkan sama yaitu 5 topik. Hal ini dilakukan agar tahap perbandingan lebih seimbang dengan membandingkan model dengan parameter yang sama.

Berdasarkan nilai *topic coherence* yang didapatkan, dihitung rata-rata nilai *topic coherence* setiap jumlah topik. Dalam menganalisis nilai *topic coherence*, semakin tinggi nilai menunjukkan hasil yang semakin baik. Rata-rata nilai *topic coherence* masing-masing skenario dapat dilihat pada Diagram 6.8.

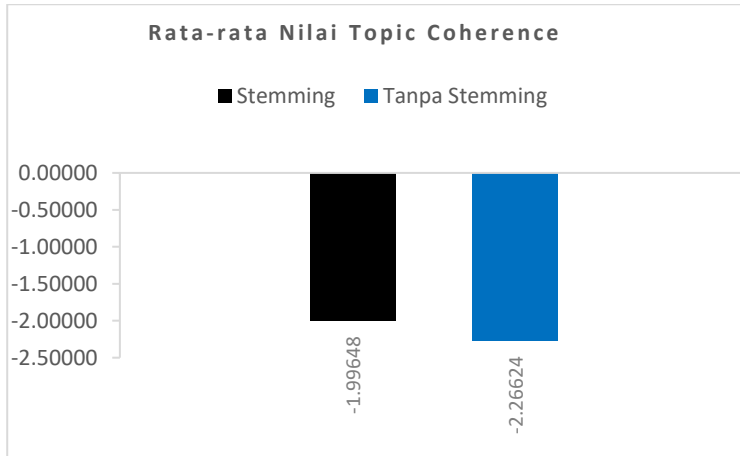


Diagram 6. 10. Rata-rata nilai *topic coherence*

Pada Diagram 6.10, diketahui bahwa model dengan skenario *stemming* memiliki rata-rata nilai *topic coherence* yang lebih tinggi, yaitu -1.99648. Visualisasi standar deviasi dari nilai *topic coherence* pada masing-masing skenario dapat dilihat pada Diagram 6.7.

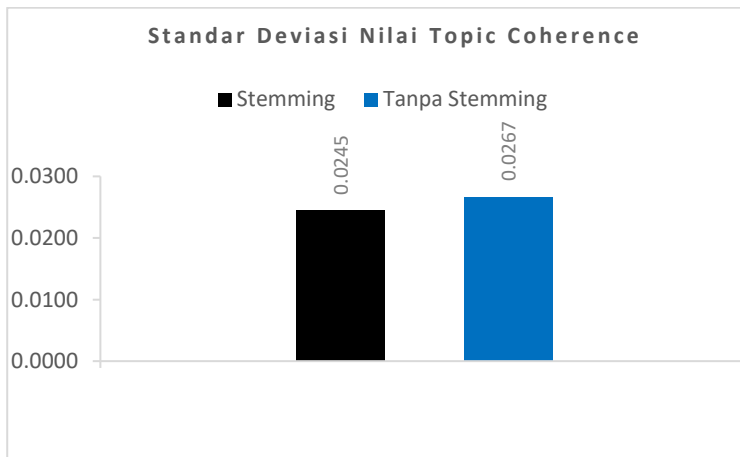


Diagram 6. 11. Standar deviasi nilai *topic coherence*

Pada Diagram 6.11, diketahui bahwa model dengan skenario *stemming* memiliki nilai standar deviasi sedikit lebih rendah, yaitu 0.0245. Dengan perbedaan 0.0022 dengan nilai

standar deviasi model skenario tanpa *stemming*, menjelaskan bahwa nilai standar deviasi *topic coherence* kedua skenario memiliki rentang tidak jauh berbeda.

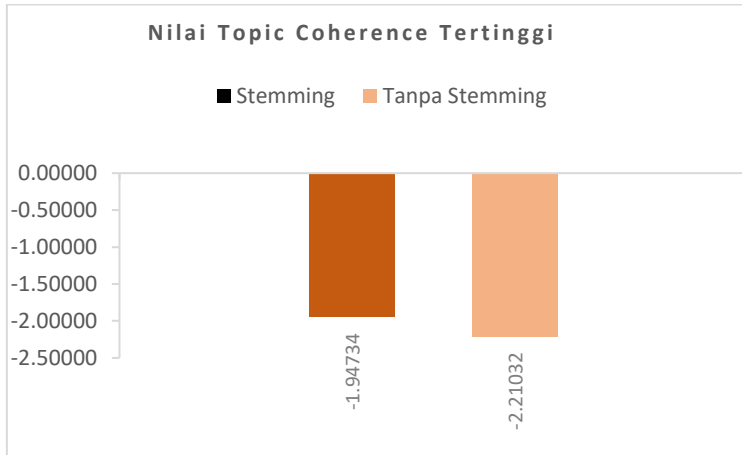


Diagram 6. 12. Nilai *topic coherence* tertinggi

Pada Diagram 6.12 diketahui bahwa skenario *stemming* memiliki nilai *topic coherence* maksimal yang lebih tinggi dari skenario tanpa *stemming*, yaitu -1.94734. Hal ini menunjukkan bahwa model pada skenario *stemming* memiliki kualitas interpretasi manusia terhadap topik yang lebih baik.

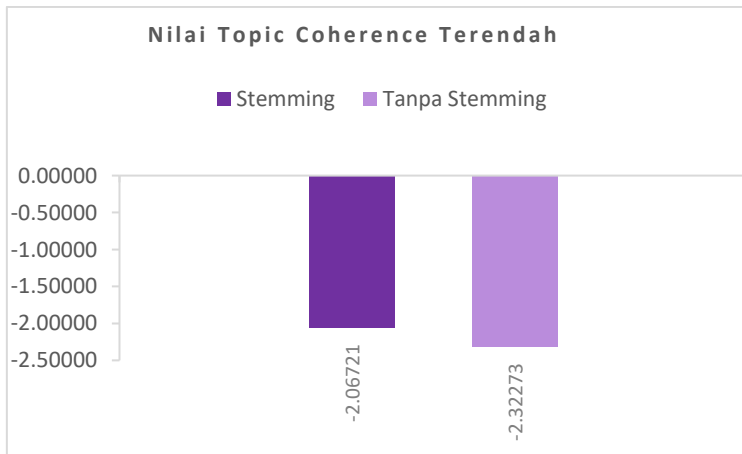


Diagram 6. 13. Nilai *topic coherence* terendah

Pada Diagram 6.13 diketahui bahwa skenario *stemming* memiliki nilai *topic coherence* minimum yang lebih tinggi dari skenario tanpa *topic*, yaitu -2.06721.

Berdasarkan diagram yang telah dijelaskan di atas, dapat disimpulkan bahwa model dengan skenario *stemming* memiliki nilai *topic coherence* yang lebih baik dibandingkan model dengan skenario tanpa *stemming*.

Dari tahap pemodelan topik berdasarkan nilai *perplexity* yang menjelaskan tingkan ketidakmiripan antar topik dan nilai *topic coherence* yang menjelaskan kualitas topik berdasarkan interpretasi manusia, didapatkan model *stemming* dengan jumlah topik 5 dengan nilai rata-rata *perplexity* yang rendah, yaitu 73.66 dan nilai *topic coherence* tertinggi yaitu -1.99648. Kedua nilai ini menjelaskan bahwa model *stemming* dengan jumlah topik 5 memiliki topik-topik yang lebih tidak mirip dan lebih sesuai dengan penilaian manusia atau *human judgement* dibandingkan model lain.

6.6 Validasi Model Topik *Gaussian LDA*

Pada validasi model topik dilakukan perhitungan *topic coherence* dari model yang dihasilkan. Hasil jumlah topik yang didapat pada masing-masing skenario sama, sehingga jumlah topik yang akan dibandingkan sama yaitu 25 topik. Hal ini dilakukan agar tahap perbandingan lebih seimbang dengan membandingkan model dengan parameter yang sama.

Perhitungan nilai *topic coherence* dilakukan dengan menggunakan *pointwise mutual information* setiap topik. Selanjutnya dihitung rata-rata nilai *PMI* dari jumlah seluruh nilai *PMI* masing-masing topik untuk mendapatkan nilai *PMI* dari model. Rata-rata nilai *topic coherence* masing-masing skenario model topik ditampilkan pada Diagram 6.12.

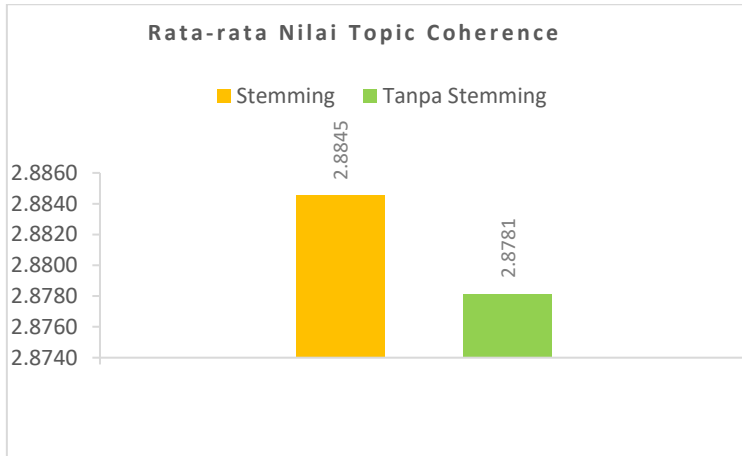


Diagram 6. 14. Rata-Rata Nilai *Topic Coherence* 25 Topik dengan *Pointwise Mutual Information*

Pada Diagram 6.14, diketahui bahwa model dengan skenario *stemming* memiliki nilai *topic coherence* dengan *pointwise mutual information* lebih tinggi, yaitu 2.8845. Standar deviasi nilai *topic coherence* masing-masing skenario topik dengan stemming ditampilkan pada Diagram 6.13.

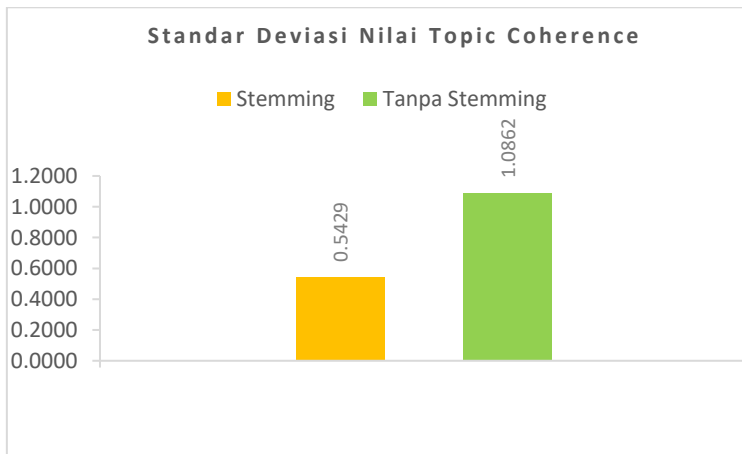


Diagram 6. 15. Standar Deviasi Nilai *Topic Coherence* 25 Topik dengan *Pointwise Mutual Information*

Pada Diagram 6.15, diketahui bahwa model dengan skenario *stemming* memiliki standar deviasi nilai *topic coherence* dengan *pointwise mutual information* lebih rendah, yaitu 0.5429. Dengan perbedaan 0.5433 dengan nilai standar deviasi model skenario tanpa *stemming*, menjelaskan bahwa nilai standar deviasi *topic coherence* kedua skenario memiliki rentang yang cukup berbeda.

Kedua nilai ini menjelaskan bahwa model *stemming* dengan jumlah topik 25 memiliki topik-topik yang lebih tidak mirip dan lebih sesuai dengan penilaian manusia atau *human judgement* dibandingkan model lain.

6.7 Pengujian Model dengan *Pointwise Mutual Information*

Pengujian model dengan menggunakan metode *pointwise mutual information* dilakukan dengan membandingkan nilai PMI dari model terbaik yang dihasilkan oleh ke dua metode yaitu *Latent Dirichlet Allocation* dan *Gaussian LDA*. Jumlah topik model terbaik dari *LDA* adalah 5 topik dengan menggunakan *stemming*, sedangkan jumlah topik model terbaik dari *Gaussian LDA* adalah 25 topik dengan menggunakan *stemming*. Untuk mendapatkan perbandingan yang seimbang, maka skenario yang dilakukan dalam pengujian ini adalah 4 skenario yaitu *Gaussian LDA* dengan 5 dan 25 topik serta *LDA* dengan 5 dan 25 topik. Pembentukan skenario ini berdasarkan jumlah topik terbaik dari masing-masing metode.

6.7.1 Analisis Kuantitatif

Analisis kuantitatif dilakukan dengan membandingkan hasil perhitungan rata-rata *pointwise mutual information* pada masing-masing model. Perhitungan PMI untuk setiap topik dilakukan dengan memilih 5 kata dengan probabilitas tertinggi pada topik tersebut. Daftar dengan probabilitas tertinggi pada masing-masing skenario ditampilkan pada Lampiran B.

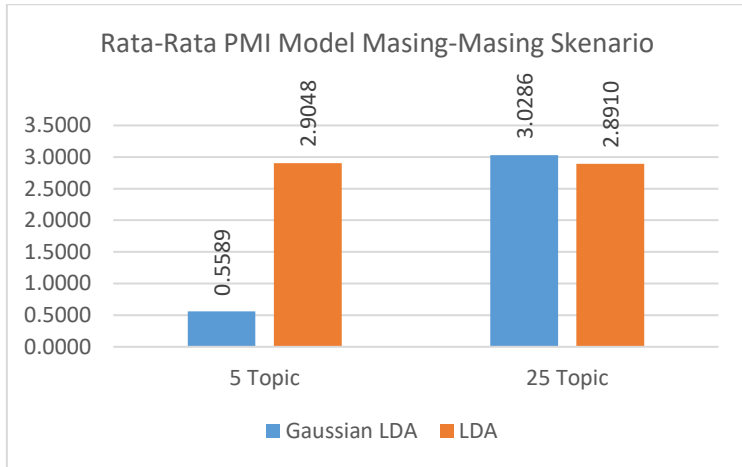


Diagram 6. 16. Nilai Rata-Rata PMI Model pada Masing-Masing Skenario

Dari diagram 6.16 didapatkan nilai PMI model pada skenario jumlah topik 5 *LDA* memiliki nilai lebih tinggi dari *Gaussian LDA*, yaitu 2.9048 dibanding 0.5589. Sebaliknya nilai PMI model pada skenario jumlah topik 25 *Gaussian LDA* memiliki nilai lebih tinggi dari *LDA*, yaitu 3.0286 dibanding 2.8910. Berdasarkan perhitungan *pointwise mutual information* yang dilakukan, disimpulkan bahwa pada jumlah topik 5 model *LDA* memiliki nilai *coherence* atau kemiripan semantik yang lebih baik dibandingkan model *Gaussian LDA*. Sebaliknya pada jumlah topik 25 model *Gaussian LDA* memiliki nilai *coherence* atau kemiripan semantik yang lebih baik dibandingkan model *LDA*.

6.7.2 Analisis Kualitatif

Analisis kualitatif dilakukan oleh konselor atau psikolog dengan membandingkan kata-kata yang memiliki probabilitas tertinggi pada masing-masing skenario. Berdasarkan tabel pada Lampiran B dapat dilihat 5 kata dengan probabilitas tertinggi pada masing-masing topik dari masing-masing model. Didapatkan bahwa model dengan menggunakan metode *Gaussian LDA* dapat mengelompokkan kata-kata

dengan kemiripan semantik yang lebih baik pada jumlah topik 25, sedangkan pada topik dengan jumlah 5 pengelompokkan yang dihasilkan tidak merata karena hasil perhitungan pada saat sampling mengindikasikan kata-kata pada corpus berada pada satu topik yang sama.

Sedangkan model yang didapatkan dari metode *LDA* kata-kata dapat dikelompokkan secara merata, baik pada jumlah topik 5 maupun 25. Namun interpretasi terhadap topik dengan jumlah 25 tidak bisa dilakukan dengan maksimal. Hal ini terjadi karena terdapat kata-kata yang memiliki kemiripan semantik yang kurang baik namun berada pada topik yang sama dengan probabilitas yang tinggi atau kata-kata yang memiliki kemiripan semantik yang baik namun berada pada topik yang berbeda. Sebagai contoh kata “pakai, agama, baca, tonton, dan mama” pada topik ke-13 memiliki interpretasi yang berbeda-beda, serta kata “pacar, sayang, dan hubung” pada topik ke-2 dan kata “cinta dan hubung” pada topik ke-3. Sedangkan kata-kata dengan probabilitas tinggi pada topik dengan jumlah 5 dirasa memiliki kemiripan semantik yang lebih baik.

Berdasarkan hasil yang didapatkan dari analisis kuantitatif dan kualitatif, masing-masing metode yang digunakan memiliki model terbaik pada jumlah topik yang berbeda. Metode *Gaussian LDA* menghasilkan model terbaik dengan jumlah topik 25, sedangkan metode *LDA* menghasilkan terbaik dengan jumlah topik 5.

6.8 Pengujian Model pada Aplikasi

Pengujian dalam penelitian ini menggunakan data testing sejumlah 214 data, yaitu data konseling pada bulan Januari dan Februari 2018. Data dilabelkan berdasarkan topik yang telah teridentifikasi sebelumnya oleh konselor. Pengujian pada aplikasi dilakukan dengan menggunakan 4 skenario yang telah didefinisikan, yaitu skenario metode *Gaussian LDA* dengan jumlah topik 5 dan 25 serta metode *LDA* dengan jumlah topik 5 dan 25.

Sebelum pengujian model pada aplikasi, dilakukan penamaan terhadap setiap topik dari masing-masing model. Penamaan topik dilakukan oleh konselor dengan melihat susunan kata dengan probabilitas tertinggi pada setiap topik.

- Model LDA
 - Jumlah Topik 5
 - Topik 0 : kuliah, orang_tua, sekolah, ajar, masuk, kelas, lulus, pilih, teman, guru. Berdasarkan kata-kata yang menyusun topik ke – 0 maka ditentukan topik 0 adalah Topik Pendidikan.
 - Topik 1 : pikir, hidup, laku, takut, teman, hati, hadap, sakit, keluarga, milik. Berdasarkan kata-kata yang menyusun topik ke – 1 maka ditentukan topik 1 adalah Topik Emosional (Kepribadian).
 - Topik 2 : pacar, hubung, cinta, sayang, putus, mantan, laki, bilang, jalan, laku. Berdasarkan kata-kata yang menyusun topik ke – 2 maka ditentukan topik 2 adalah Topik Cinta.
 - Topik 3 : anak, rumah, suami, ayah, keluarga, orang_tua, kakak, uang, adik, tinggal. Berdasarkan kata-kata yang menyusun topik ke – 3 maka ditentukan topik 3 adalah Topik Keluarga.
 - Topik 4 : teman, suka, bicara, cewek, bilang, cowok, sahabat, kelas, lihat, kaya. Berdasarkan kata-kata yang menyusun topik ke – 4 maka ditentukan topik 4 adalah Topik Sosial.
 - Jumlah Topik 25
 - Topik 0 : hidup, sakit, bahagia, sedih hati. Berdasarkan kata-kata yang menyusun topik ke – 0 maka ditentukan topik 0 adalah Topik Perasaan.
 - Topik 1 : laki, hubung, pasang, jalan, perempuan. Berdasarkan kata-kata yang menyusun topik ke – 1 maka ditentukan topik 1 adalah Topik Pasangan.

- Topik 2 : pacar, putus, saying, mantan, hubung. Berdasarkan kata-kata yang menyusun topik ke – 2 maka ditentukan topik 2 adalah Topik Hubungan.
- Topik 3 : cinta, temu, wanita, hubung, asa. Berdasarkan kata-kata yang menyusun topik ke – 3 maka ditentukan topik 3 adalah Topik Wujud.
- Topik 4 : bicara, malu, laku, nyaman, jelek. Berdasarkan kata-kata yang menyusun topik ke – 4 maka ditentukan topik 4 adalah Topik Sifat.
- Topik 5 : sekolah, ajar, orang_tua, guru, kuliah. Berdasarkan kata-kata yang menyusun topik ke – 5 maka ditentukan topik 5 adalah Topik Belajar.
- Topik 6 : keluarga, orang_tua, ayah, anak, tinggal. Berdasarkan kata-kata yang menyusun topik ke – 6 maka ditentukan topik 6 adalah Topik Keluarga.
- Topik 7 : sosial, sepi, media, benci, sibuk. Berdasarkan kata-kata yang menyusun topik ke – 7 maka ditentukan topik 7 adalah Topik Sifat.
- Topik 8 : suka, asa, hati, kadang, curah. Berdasarkan kata-kata yang menyusun topik ke – 8 maka ditentukan topik 8 adalah Topik Keadaan.
- Topik 9 : marah, ubah, sikap, diam, sifat. Berdasarkan kata-kata yang menyusun topik ke – 9 maka ditentukan topik 9 adalah Topik Perilaku.
- Topik 10 : paham, tertawa, hina, mood, sulit. Berdasarkan kata-kata yang menyusun topik ke – 10 maka ditentukan topik 10 adalah Topik Reaksi.
- Topik 11 : suami, anak, menikah, nikah, istri. Berdasarkan kata-kata yang menyusun topik ke – 11 maka ditentukan topik 11 adalah Topik Status.
- Topik 12 : teman, cerita, butuh, anggap, dengar. Berdasarkan kata-kata yang menyusun topik ke – 12 maka ditentukan topik 12 adalah Topik Kebutuhan.

- Topik 13 : pakai, agama, baca, tonton, mama. Berdasarkan kata-kata yang menyusun topik ke – 13 maka ditentukan topik 13 adalah Topik Sistem / Aturan.
- Topik 14 : cewek, cowok, suka, lihat, perhatian. Berdasarkan kata-kata yang menyusun topik ke – 14 maka ditentukan topik 14 adalah Topik Jenis.
- Topik 15 : kuliah, kampus, jurusan, masuk, tugas. Berdasarkan kata-kata yang menyusun topik ke – 15 maka ditentukan topik 15 adalah Topik Pendidikan.
- Topik 16 : takut, piker, tidur, hilang, laku. Berdasarkan kata-kata yang menyusun topik ke – 16 maka ditentukan topik 16 adalah Topik Situasi.
- Topik 17 : rumah, pulang, kota, izin, pergi. Berdasarkan kata-kata yang menyusun topik ke – 17 maka ditentukan topik 17 adalah Topik Waktu.
- Topik 18 : teman, kelas, bicara, suka, main. Berdasarkan kata-kata yang menyusun topik ke – 18 maka ditentukan topik 18 adalah Topik Tingkat.
- Topik 19 : sahabat, biar, mimpi, percaya, bikin. Berdasarkan kata-kata yang menyusun topik ke – 19 maka ditentukan topik 19 adalah Topik Pengaruh.
- Topik 20 : bilang, ketemu, balas, maaf, pesan. Berdasarkan kata-kata yang menyusun topik ke – 20 maka ditentukan topik 20 adalah Topik Ungkapan.
- Topik 21 : bantu, tolong, saran, mohon, solusi. Berdasarkan kata-kata yang menyusun topik ke – 21 maka ditentukan topik 21 adalah Topik Permintaan.
- Topik 22 : uang, butuh, gara, makan, capek. Berdasarkan kata-kata yang menyusun topik ke – 22 maka ditentukan topik 22 adalah Topik Hasil.

- Topik 23 : kakak, adik, mati, saudara, tanda. Berdasarkan kata-kata yang menyusun topik ke – 23 maka ditentukan topik 23 adalah Topik Sapaan.
 - Topik 24 : pilih, bingung, usaha, cari, lulus. Berdasarkan kata-kata yang menyusun topik ke – 24 maka ditentukan topik 24 adalah Topik Penjurusan / Seleksi.
- Model Gaussian LDA
 - Jumlah Topik 5
 - Topik 0 : - .
 - Topik 1 : - .
 - Topik 2 : - .
 - Topik 3 : - .
 - Topik 4 : peduli, teman, nyaman, jujur, jahat. Penentuan nama topik tidak dapat dilakukan karena seluruh kata pada corpus terkelompokkan pada topik 4.
 - Jumlah Topik 25
 - Topic 0 : uang, duit, bayar, juta, jajan, modal, ribu, hutang, biaya, pinjam, barang. Berdasarkan kata-kata yang menyusun topik ke – 0 maka ditentukan topik 0 adalah Topik Keuangan.
 - Topic 1 : pulang, jemput, kantin, jalan, makan, bareng, belanja, parkir, pamit, pergi, berangkat. Berdasarkan kata-kata yang menyusun topik ke – 1 maka ditentukan topik 1 adalah Topik Kegiatan.
 - Topic 2 : optimis, pesimis, lambat, ubah, semangat, turun, jernih, bangkit, motivasi, efektif, konsen. Berdasarkan kata-kata yang menyusun topik ke – 2 maka ditentukan topik 2 adalah Topik Motivasi.
 - Topic 3 : kasih, tipu, kasi, sali, tabung, tuntutan. Berdasarkan kata-kata yang menyusun topik ke – 3 maka ditentukan topik 3 adalah Topik Pekerjaan.

- Topic 4 : teman, cowok, cewek, sahabat, pacar, perempuan, lelaki, laki, orang, mantan, pria. Berdasarkan kata-kata yang menyusun topik ke – 4 maka ditentukan topik 4 adalah Topik Relasi / Pertemanan.
- Topic 5 : takdir, rangking, juara, syukur, tuhan, doa, peringkat, lolos, enam, nihil, ikhlas. Berdasarkan kata-kata yang menyusun topik ke – 5 maka ditentukan topik 5 adalah Topik Ketetapan.
- Topic 6 : psikis, perilaku, psikologis, gejala, dampak, mental, disorder, alami, karakter, emosional, bipolar. Berdasarkan kata-kata yang menyusun topik ke – 6 maka ditentukan topik 6 adalah Topik Reaksi.
- Topic 7 : sayang, jujur, kecewa, ragu, serius, percaya, benci, curiga, sadar, penasaran, cemburu. Berdasarkan kata-kata yang menyusun topik ke – 7 maka ditentukan topik 7 adalah Topik Perasaan / Cinta.
- Topic : bunyi, celana, kiri, kanan, kencang, kaca, basah, perut, leher, darah, telinga. Berdasarkan kata-kata yang menyusun topik ke – 8 maka ditentukan topik 8 adalah Topik Fisik.
- Topic 9 : kuliah, sekolah, smk, lulus, masuk, ipa, pesantren, pindah, tamat, bimbel, swasta. Berdasarkan kata-kata yang menyusun topik ke – 9 maka ditentukan topik 9 adalah Topik Pendidikan.
- Topic : ayah, mertua, adik, keluarga, nenek, suami, anak, tua, mama, saudara, kandung. Berdasarkan kata-kata yang menyusun topik ke – 10 maka ditentukan topik 10 adalah Topik Keluarga.
- Topic 11 : suka, bawel, cuek, kalem, sinis, risih, polos, egois, cantik, sebal, jahat. Berdasarkan kata-kata yang menyusun topik ke – 11 maka ditentukan topik 11 adalah Topik Sifat.

- Topic 12 : hidup, hati, bahagia, benak, nyata, dunia, mustahil, utuh, pendam, hadir, rindu. Berdasarkan kata-kata yang menyusun topik ke – 12 maka ditentukan topik 12 adalah Topik Keadaan.
- Topic 13 : usaha, rezeki, gelar, rencana, target, syukur, beasiswa, prinsip, modal, tujuan, hak. Berdasarkan kata-kata yang menyusun topik ke – 13 maka ditentukan topik 13 adalah Topik Tujuan.
- Topic 14 : bicara, bilang, dibilangin, bilangan, ngeluh, ganggu, marah, digituin, diomelin, sindir, bentak. Berdasarkan kata-kata yang menyusun topik ke – 14 maka ditentukan topik 14 adalah Topik Pengungkapan / Perasaan.
- Topic 15 : nilai, fisika, prestasi, matematika, ipk, tugas, les, peringkat, presentasi, praktek, guru. Berdasarkan kata-kata yang menyusun topik ke – 15 maka ditentukan topik 15 adalah Topik Presatasi / Tingkat.
- Topic 16 : kaku, susah, malas, sulit, mudah, jarang, kadang, gampang, enggan, gugup, minder. Berdasarkan kata-kata yang menyusun topik ke – 16 maka ditentukan topik 16 adalah Topik Karakter / Kepribadian.
- Topic 17 : saran, solusi, terima, nasehat, nasihat, motivasi, tolong, kado, maaf, suport, bantu. Berdasarkan kata-kata yang menyusun topik ke – 17 maka ditentukan topik 17 adalah Topik Pertolongan.
- Topic 18 : kelas, smp, klas, sd, bangku, smk, beda, kenal, dulunya, murid, sekolah. Berdasarkan kata-kata yang menyusun topik ke – 18 maka ditentukan topik 18 adalah Topik Tingkatan.
- Topic 19 : teknik, akuntansi, fakultas, psikologi, desain, manajemen, hukum, daftar, sarjana, ips, ptn. Berdasarkan kata-kata yang menyusun topik

ke – 19 maka ditentukan topik 19 adalah Topik Penjurusan.

- Topic 20 : rumah, kos, asrama, kota, jakarta, kampung, kost, hotel, daerah, yogyakarta, tinggal. Berdasarkan kata-kata yang menyusun topik ke – 20 maka ditentukan topik 20 adalah Topik Tempat.
- Topic 21 : ig, dm, instagram, bbm, bm, instagramnya, profil, snapgram, kirim, foto, blokir. Berdasarkan kata-kata yang menyusun topik ke – 21 maka ditentukan topik 21 adalah Topik Sosial Media.
- Topic 22 : sedih, sakit, lelah, kacau, stres, tenang, gelisah, frustrasi, jenuh, muak, berat. Berdasarkan kata-kata yang menyusun topik ke – 22 maka ditentukan topik 22 adalah Topik Emosional.
- Topic 23 : menikah, nikah, umur, usia, hamil, cerai, muda, kawin, perawan, mapan, duda. Berdasarkan kata-kata yang menyusun topik ke – 23 maka ditentukan topik 23 adalah Topik Pernikahan.
- Topic 24 : sabtu, jumat, sore, malam, lebaran, libur, jam, pagi, senin, kamsis, tunggu. Berdasarkan kata-kata yang menyusun topik ke – 24 maka ditentukan topik 24 adalah Topik Waktu.

Selanjutnya dilakukan pengujian hasil prediksi pada aplikasi dengan label yang diberikan oleh konselor pada masing-masing data testing dengan label dari 3 jenis skenario yang didapatkan. Skenario yang dimaksud adalah *Gaussian LDA* dengan jumlah topik 25 serta *LDA* dengan jumlah topik 5 dan 25. Sedangkan hasil model *Gaussian LDA* dengan jumlah topik 5 tidak dapat digunakan karena seluruh kata pada *corpus* terkelompokkan hanya pada 1 topik yang sama.

Dari hasil dari pengujian hasil prediksi aplikasi dengan label yang diberikan, dihitung prosentasi akurasi yang didapat. Perhitungan dilakukan dengan rumus :

$$akurasi = \frac{jumlah\ data\ tepat\ berlabel}{total\ seluruh\ data}$$

Data dikatakan akurat pada jumlah topik 5 apabila label yang diberikan oleh konselor sama dengan topik dengan probabilitas tertinggi yang diberikan oleh aplikasi. Sedangkan pada jumlah topik 25, dikatakan akurat apabila label yang diberikan oleh konselor berada pada 5 topik dengan probabilitas tertinggi yang diberikan oleh aplikasi. Hasil dari pengujian terhadap masing-masing model skenario dalam aplikasi ditunjukkan pada Lampiran C.

Pada jumlah topik 5, metode *Gaussian LDA* tidak dapat dilakukan pengujian pada jumlah topik ini karena seluruh data *corpus* terkelompokkan menjadi 1 kelompok yang sama dengan probabilitas 1.0. Sedangkan metode *Latent Dirichlet Allocation* menghasilkan data yang terlabelkan dengan tepat dengan jumlah 161 data dan data yang memiliki perbedaan label dari yang seharusnya adalah 53 data. Sehingga kemudian didapatkan akurasi yaitu 161/214 atau sama dengan 75.2% serta tingkat kesalahan atau *error rate* dari pengujian skenario ini 53/214 atau sama dengan 24.8%.

Pada jumlah topik 25, metode *Gaussian LDA* menghasilkan data yang terlabelkan dengan tepat dengan jumlah 173 data dan data yang memiliki perbedaan label dari yang seharusnya atau tidak termasuk 5 topik tertinggi yang dihasilkan adalah 41 data. Sehingga kemudian didapatkan akurasi yaitu 173/214 atau sama dengan 80.84% serta tingkat kesalahan atau *error rate* dari pengujian skenario ini 41/214 atau sama dengan 19.16 %. Sedangkan metode *Latent Dirichlet Allocation* menghasilkan data yang terlabelkan dengan tepat dengan jumlah 18 data dan data yang memiliki perbedaan label dari yang seharusnya atau tidak termasuk 5 topik tertinggi yang dihasilkan adalah 196 data. Sehingga kemudian didapatkan akurasi yaitu 18/214 atau sama dengan 8.41% serta tingkat kesalahan atau *error rate* dari pengujian skenario ini 196/214 atau sama dengan 91.59%.

Berdasarkan pengujian yang dilakukan, didapatkan pengujian pada skenario jumlah topik 5 model metode *LDA* memiliki akurasi lebih tinggi dibandingkan model metode *Gaussian LDA*, yaitu sebesar 75.2%. Sebaliknya pada skenario jumlah topik 25 model metode *Gaussian LDA* memiliki akurasi lebih tinggi dibandingkan model metode *LDA*, yaitu sebesar 80.84%. Perbandingan nilai akurasi pada masing-masing jumlah topik selaras dengan perbandingan nilai *coherence* model dengan perhitungan PMI, dimana pada jumlah topik 5 model metode *LDA* memiliki nilai *coherence* lebih baik dan sebaliknya pada jumlah topik 25 model metode *Gaussian LDA* memiliki nilai *coherence* yang lebih baik.

Selanjutnya dilakukan pemilihan model yang akan digunakan sebagai input implementasi aplikasi dalam pendeteksian topik terhadap konseling baru. Pemilihan model dilakukan dengan diskusi antara Riliv, konselor, dan peneliti. Hasil yang didapat, dipilih model *LDA* dengan jumlah topik 5 dan model *Gaussian LDA* dengan jumlah model 25. Kedua model dipilih karena dinilai dapat menghasilkan topik dengan kemiripan semantik yang baik pada masing-masing jumlah topik. Pertimbangan lain adalah model *LDA* dengan jumlah topik 5 dapat memberikan interpretasi topik awal yang baik kepada psikolog, sedangkan model *Gaussian LDA* dengan jumlah topik 25 dapat memberikan interpretasi lanjut yang lebih terperinci kepada psikolog.

Halaman Sengaja Dikosongkan

BAB VII

KESIMPULAN DAN SARAN

Pada bab ini dibahas mengenai kesimpulan dari semua proses yang telah dilakukan dan saran yang dapat diberikan untuk pengembangan yang lebih baik.

7.1 Kesimpulan

Kesimpulan yang didapatkan dari proses pengerjaan tugas akhir yang telah dilakukan antara lain :

1. Berdasarkan hasil eksperimen pengukuran masing-masing nilai *perplexity* untuk kedua metode, didapatkan model LDA dengan jumlah topik 5 topik merupakan model dengan jumlah terbaik dengan metode LDA. Sedangkan model Gaussian LDA dengan jumlah topik 25 merupakan model dengan jumlah terbaik dengan metode Gaussian LDA. Dari hasil tersebut dipilih model terbaik yaitu model *LDA stemming* dengan *passes* sejumlah 50 kali dan topik sejumlah 5 serta model *Gaussian LDA stemming* dengan iterasi 2 kali dan topik sejumlah 25.
2. Perhitungan *Pointwise Mutual Information* menghasilkan pada jumlah topik 5 nilai *coherence* model *LDA stemming* lebih baik dibandingkan model *Gaussian LDA stemming*. Sebaliknya pada jumlah topik 25 nilai *coherence* model *Gaussian LDA stemming* lebih baik dibandingkan model *LDA stemming*.
3. Pengujian model pada aplikasi dilakukan dengan menggunakan model terbaik pada masing-masing jumlah topik. Hasil yang didapat pada jumlah topik 5 dengan menggunakan model metode *LDA* menghasilkan nilai akurasi sebesar 75.2 %. Kemudian hasil yang didapat pada jumlah topik 25 dengan menggunakan model metode *Gaussian LDA* menghasilkan nilai akurasi sebesar 80.81%.

7.2 Saran dan Penelitian Selanjutnya

Dari pengerjaan tugas akhir ini, adapun beberapa saran untuk pengembangan penelitian ke depan.

1. Pada proses *formalizer* pada tahap pra-proses, kamus data *formalizer* dari *repository* yang telah disusun Dr. Eng. Ayu Purwarianti, ST.,MT., et al [15] dan kamus data kata baku Bahasa Indonesia masih belum dapat mencakup keseluruhan kata pada *corpus*. Sehingga diperlukan perbendaharaan kata yang lebih banyak dan lebih baru untuk kamus data *formalizer* dan kamus data kata baku Bahasa Indonesia.
2. Pada proses *stemming* pada tahap pra-proses menggunakan *library Sastrawi* masih terdapat kata-kata yang memiliki hasil tidak sesuai dengan seharusnya. Sehingga untuk mengoptimalkan hasil *stemming* diperlukan *library* lain yang memiliki hasil *stemming* lebih baik.
3. Berdasarkan hasil pemodelan topik didapatkan masih banyak kata yang belum tercakup pada model *word2vec* yang dikembangkan oleh Dr. Eng. Ayu Purwarianti, ST.,MT., et al, sehingga perlu dikembangkan lebih lanjut atau diperlukan penggunaan model *word2vec* hasil pengembangan yang lebih baik.
4. Penentuan jumlah topik pada Gaussian LDA memerlukan metode lain sebagai pembanding penentuan jumlah topik menggunakan perbandingan nilai *perplexity* model agar hasil yang didapat dipercaya, sehingga diperlukan pencarian metode penentuan jumlah topik lain dengan studi literatur.
5. Proses pemberian label masih dilakukan oleh mahasiswa Psikologi masih belum memiliki tingkat interpretasi topik model yang tinggi karena belum memiliki pengalaman yang cukup dalam melakukan konseling sosial. Sehingga diperlukan psikolog yang telah memiliki pengalaman yang cukup dan kemampuan lebih dalam melakukan interpretasi topik hasil pemodelan.

DAFTAR PUSTAKA

- [1] “Riliv - Curhat Online dan Konseling dengan Psikolog.” [Online]. Available: <https://riliv.co/>. [Accessed: 04-Jan-2018].
- [2] D. M. Blei, B. B. Edu, A. Y. Ng, A. S. Edu, M. I. Jordan, and J. B. Edu, “Latent Dirichlet Allocation,” *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, 2003.
- [3] R. Das, M. Zaheer, and C. Dyer, “Gaussian LDA for Topic Models with Word Embeddings,” *Proc. ACL 2015*, pp. 795–804, 2015.
- [4] D. M. Blei, “Probabilistic topic models,” *Commun. ACM*, vol. 55, no. 4, p. 77, Apr. 2012.
- [5] Megan R. Brett, “Topic Modeling: A Basic Introduction,” *Communications of the ACM*. [Online]. Available: <http://journalofdigitalhumanities.org/2-1/topic-modeling-a-basic-introduction-by-megan-r-brett/>. [Accessed: 10-Feb-2018].
- [6] J. C. Campbell, A. Hindle, and E. Stroulia, “Latent Dirichlet Allocation : Extracting Topics from Software Engineering Data,” pp. 1–21, 2014.
- [7] A. Rajaraman and J. D. Ullman, “Data Mining,” *Min. Massive Datasets*, vol. 18 Suppl, pp. 114–142, 2011.
- [8] J. Ramos, “Using TF-IDF to Determine Word Relevance in Document Queries,” *Proc. first Instr. Conf. Mach. Learn.*, pp. 1–4, 2003.
- [9] K. Stevens, P. Kegelmeyer, D. Andrzejewski, and D. Buttler, “Exploring Topic Coherence over Many Models and Many Topics,” *Proc. 2012 Jt. Conf. Empir. Methods Nat. Lang. Process. Comput. Nat. Lang. Learn.*, no. July, pp. 952–961, 2012.

- [10] M. Steyvers and T. Griffiths, “Probabilistic Topic Models,” pp. 1–15.
- [11] J. Chang, S. Gerrish, C. Wang, and D. M. Blei, “Reading Tea Leaves: How Humans Interpret Topic Models,” *Adv. Neural Inf. Process. Syst.* 22, pp. 288–296, 2009.
- [12] K. W. Church and P. Hanks, “Word association norms, mutual information, and lexicography,” *Proc. 27th Annu. Meet. Assoc. Comput. Linguist.* - , vol. 16, no. 1, pp. 76–83, 1989.
- [13] D. Newman, Y. Noh, E. Talley, S. Karimi, and T. Baldwin, “Evaluating topic models for digital libraries,” *Proc. 10th ACM/IEEE-CS Jt. Conf. Digit. Libr.*, no. January, pp. 215–224, 2010.
- [14] “Why Laravel is best PHP Framework.” [Online]. Available: <https://blog.vanila.io/why-laravel-is-best-php-framework-98a2784d76dc>. [Accessed: 20-Feb-2018].
- [15] A. Purwarianti, A. Andhika, A. F. Wicaksono, I. Afif, and F. Ferdian, “InaNLP: Indonesia natural language processing toolkit, case study: Complaint tweet classification,” in *2016 International Conference On Advanced Informatics: Concepts, Theory And Application (ICAICTA)*, 2016, pp. 1–5.
- [16] “Pujangga - Indonesian Natural Language Processing REST API, an Interface for InaNLP and Deeplearning4j’s Word2Vec.” [Online]. Available: <https://github.com/panggi/pujangga>. [Accessed: 04-Jan-2018].

BIODATA PENULIS



Penulis lahir di Ponorogo pada tanggal 20 April 1996. Penulis merupakan anak kedua dari 3 bersaudara. Penulis telah menempuh beberapa pendidikan formal yaitu, SD Negeri Kemuning, SMP Negeri 1 Ponorogo, dan SMA Negeri 1 Ponorogo.

Pada tahun 2014 pasca kelulusan SMA, penulis melanjutkan pendidikan dengan jalur SNMPTN di Departemen Sistem Informasi Fakultas Teknologi Informasi dan Komunikasi - Institut Teknologi Sepuluh Nopember (ITS) Surabaya dan terdaftar sebagai mahasiswa dengan Nomor Registrasi Pokok (NRP) 5214100080 / 05211440000080. Selama menjadi mahasiswa, penulis mengikuti berbagai kegiatan kemahasiswaan seperti beberapa kepanitiaan serta pernah menjabat sebagai Staff Departemen Aplikasi Teknologi HMSI ITS 2015/2016, Staff Departemen Internal Affairs BEM FTIf ITS 2015/2016, dan Ketua Departemen Internal Affairs BEM FTIf ITS 2016/2017. Selain itu, penulis telah mengikuti Latihan Keterampilan Manajemen Mahasiswa (LKMM) mulai dari Tingkat Pra-Dasar, Tingkat Dasar, Tingkat Menengah, serta Pelatihan Pemandu LKMM ITS. Di bidang akademik, penulis aktif sebagai asisten dosen dan asisten praktikum pada beberapa mata kuliah seperti Pemrograman Berbasis Website, Desain Basis Data, dan Sistem Cerdas.

Pada tahun keempat, karena penulis memiliki keterampilan di bidang pengolahan data, maka penulis mengambil bidang minat Akuisisi Data dan Diseminasi Informasi (ADDI). Penulis dapat dihubungi melalui *email* di anugrahdputra@gmail.com

LAMPIRAN A

Tabel Hasil Eksperimen Penentuan Jumlah *Passes Latent Dirichlet Allocation*

Iterasi	Jumlah Topik									
	2	4	6	8	10	12	14	16	18	20
1	102.7	102.6	105.5	106.6	107.0	106.3	109.9	109.7	110.1	113.2
2	101.4	97.8	101.5	102.0	102.8	102.0	106.2	104.8	105.0	107.9
3	101.1	96.4	99.8	100.4	101.4	100.7	105.1	103.2	103.3	106.3
4	100.9	95.7	98.5	99.7	100.7	100.2	104.4	102.6	102.5	105.6
5	100.7	95.3	98.0	99.4	100.3	100.0	104.1	102.3	102.2	105.3
6	100.6	95.1	97.8	99.3	100.1	99.8	103.7	102.1	102.0	105.2
7	100.6	94.9	97.1	99.1	99.9	99.7	103.6	102.0	101.9	105.2
8	100.5	94.8	96.6	99.0	99.7	99.7	103.4	101.9	101.8	105.1
9	100.5	94.7	96.3	99.0	99.6	99.6	103.4	101.8	101.8	105.0
10	100.5	94.7	96.1	99.0	99.5	99.6	103.3	101.8	101.7	104.9
11	100.5	94.6	96.0	98.9	99.5	99.6	103.3	101.7	101.6	104.8
12	100.4	94.5	96.0	98.9	99.4	99.5	103.2	101.6	101.6	104.8
13	100.4	94.5	96.0	98.9	99.4	99.5	103.2	101.7	101.6	104.8
14	100.4	94.5	95.9	98.8	99.4	99.5	103.2	101.6	101.5	104.7
15	100.4	94.4	95.9	98.8	99.4	99.4	103.1	101.6	101.6	104.7
16	100.3	94.4	95.9	98.8	99.3	99.4	103.1	101.6	101.5	104.6
17	100.3	94.4	95.9	98.7	99.3	99.4	103.0	101.5	101.5	104.6
18	100.3	94.3	95.8	98.7	99.3	99.3	103.0	101.4	101.5	104.6

19	100.3	94.3	95.8	98.7	99.3	99.3	103.0	101.3	101.5	104.5
20	100.3	94.3	95.8	98.7	99.2	99.3	103.0	101.3	101.5	104.6
21	100.3	94.3	95.7	98.6	99.2	99.3	103.0	101.3	101.5	104.5
22	100.3	94.2	95.7	98.6	99.2	99.2	102.9	101.2	101.4	104.5
23	100.2	94.2	95.7	98.6	99.2	99.2	102.9	101.2	101.5	104.5
24	100.2	94.2	95.6	98.6	99.2	99.2	102.9	101.2	101.4	104.5
25	100.2	94.1	95.6	98.6	99.1	99.2	102.9	101.2	101.4	104.5
26	100.2	94.1	95.6	98.6	99.1	99.2	102.9	101.2	101.4	104.4
27	100.2	94.1	95.6	98.5	99.1	99.2	102.8	101.1	101.4	104.4
28	100.2	94.1	95.6	98.5	99.1	99.1	102.8	101.1	101.4	104.4
29	100.2	94.1	95.6	98.5	99.1	99.1	102.8	101.1	101.3	104.4
30	100.1	94.0	95.5	98.5	99.0	99.1	102.8	101.1	101.3	104.4
31	100.1	94.0	95.5	98.5	99.0	99.1	102.8	101.1	101.3	104.4
32	100.1	94.0	95.5	98.5	99.0	99.1	102.8	101.1	101.3	104.3
33	100.1	94.0	95.5	98.5	99.0	99.1	102.8	101.0	101.3	104.3
34	100.1	94.0	95.5	98.5	99.0	99.1	102.8	101.0	101.3	104.3
35	100.1	93.9	95.5	98.4	99.0	99.1	102.7	101.0	101.3	104.3
36	100.1	93.9	95.5	98.4	98.9	99.1	102.8	101.0	101.3	104.3
37	100.1	93.9	95.5	98.4	98.9	99.1	102.7	101.0	101.3	104.2
38	100.1	93.9	95.5	98.4	98.9	99.0	102.7	101.0	101.3	104.3
39	100.0	93.9	95.4	98.4	98.9	99.0	102.7	101.0	101.3	104.2
40	100.0	93.9	95.4	98.4	98.9	99.0	102.7	101.0	101.2	104.2
41	100.0	93.9	95.4	98.4	98.9	99.0	102.7	100.9	101.2	104.2

42	100.0	93.9	95.4	98.4	98.9	99.0	102.7	100.9	101.2	104.2
43	100.0	93.8	95.4	98.4	98.9	99.0	102.7	100.9	101.2	104.2
44	100.0	93.8	95.4	98.3	98.8	99.0	102.7	100.9	101.2	104.2
45	100.0	93.8	95.4	98.3	98.8	99.0	102.7	100.9	101.2	104.1
46	100.0	93.8	95.4	98.3	98.8	99.0	102.6	100.9	101.2	104.1
47	100.0	93.8	95.4	98.3	98.8	99.0	102.6	100.9	101.2	104.1
48	100.0	93.8	95.4	98.3	98.8	98.9	102.6	100.9	101.2	104.1
49	99.9	93.8	95.3	98.3	98.8	98.9	102.6	100.8	101.2	104.1
50	99.9	93.8	95.3	98.3	98.8	98.9	102.6	100.9	101.2	104.1
51	99.9	93.8	95.3	98.3	98.8	98.9	102.6	100.8	101.1	104.1
52	99.9	93.8	95.3	98.3	98.7	98.9	102.6	100.8	101.1	104.0
53	99.9	93.7	95.3	98.3	98.7	98.9	102.6	100.8	101.1	104.1
54	99.9	93.7	95.3	98.2	98.7	98.9	102.6	100.8	101.1	104.0
55	99.9	93.7	95.3	98.2	98.7	98.9	102.6	100.8	101.1	104.0
56	99.9	93.7	95.3	98.2	98.7	98.9	102.6	100.8	101.1	104.0
57	99.9	93.7	95.3	98.2	98.7	98.9	102.6	100.8	101.1	104.0
58	99.9	93.7	95.3	98.2	98.7	98.9	102.6	100.8	101.1	104.0
59	99.9	93.7	95.3	98.2	98.7	98.8	102.5	100.8	101.1	104.0
60	99.9	93.7	95.3	98.2	98.7	98.8	102.5	100.8	101.1	104.0
61	99.8	93.7	95.3	98.2	98.7	98.8	102.5	100.8	101.1	104.0
62	99.8	93.7	95.3	98.2	98.7	98.8	102.5	100.8	101.1	104.0
63	99.8	93.7	95.2	98.2	98.6	98.8	102.5	100.8	101.1	104.0
64	99.8	93.6	95.2	98.1	98.6	98.8	102.5	100.8	101.1	104.0

65	99.8	93.6	95.2	98.1	98.6	98.8	102.5	100.7	101.1	104.0
66	99.8	93.6	95.2	98.1	98.6	98.8	102.5	100.7	101.1	103.9
67	99.8	93.6	95.2	98.1	98.6	98.8	102.5	100.7	101.1	103.9
68	99.8	93.6	95.2	98.1	98.6	98.8	102.4	100.7	101.0	103.9
69	99.8	93.6	95.2	98.1	98.6	98.8	102.5	100.7	101.0	103.9
70	99.8	93.6	95.2	98.1	98.6	98.8	102.4	100.7	101.0	103.9
71	99.8	93.6	95.2	98.1	98.6	98.7	102.4	100.7	101.0	103.9
72	99.8	93.5	95.2	98.1	98.6	98.8	102.4	100.7	101.0	103.9
73	99.8	93.5	95.2	98.1	98.6	98.7	102.4	100.7	101.0	103.9
74	99.8	93.5	95.2	98.1	98.6	98.7	102.4	100.7	101.0	103.9
75	99.8	93.5	95.2	98.1	98.6	98.7	102.4	100.7	101.0	103.8
76	99.7	93.5	95.2	98.1	98.6	98.7	102.4	100.7	101.0	103.9
77	99.7	93.5	95.2	98.1	98.5	98.7	102.4	100.7	101.0	103.8
78	99.7	93.5	95.1	98.0	98.5	98.7	102.4	100.7	101.0	103.8
79	99.7	93.5	95.1	98.0	98.5	98.7	102.4	100.7	101.0	103.8
80	99.7	93.5	95.1	98.0	98.5	98.7	102.4	100.6	101.0	103.8
81	99.7	93.5	95.1	98.0	98.5	98.7	102.4	100.7	100.9	103.8
82	99.7	93.5	95.1	98.0	98.5	98.7	102.4	100.6	101.0	103.8
83	99.7	93.5	95.1	98.0	98.5	98.7	102.3	100.6	100.9	103.8
84	99.7	93.5	95.1	98.0	98.5	98.7	102.3	100.6	100.9	103.8
85	99.7	93.5	95.1	98.0	98.5	98.7	102.3	100.6	100.9	103.8
86	99.7	93.4	95.1	98.0	98.5	98.7	102.3	100.6	100.9	103.8
87	99.7	93.4	95.1	98.0	98.5	98.6	102.3	100.6	100.9	103.7

88	99.7	93.4	95.1	98.0	98.5	98.7	102.3	100.6	100.9	103.8
89	99.7	93.4	95.1	98.0	98.5	98.6	102.3	100.6	100.9	103.7
90	99.7	93.4	95.1	98.0	98.5	98.7	102.3	100.6	100.9	103.8
91	99.7	93.4	95.1	98.0	98.5	98.6	102.3	100.6	100.9	103.7
92	99.7	93.4	95.1	98.0	98.5	98.6	102.3	100.6	100.9	103.7
93	99.7	93.4	95.1	98.0	98.5	98.6	102.3	100.6	100.9	103.7
94	99.7	93.4	95.1	97.9	98.4	98.6	102.3	100.6	100.9	103.7
95	99.6	93.4	95.1	97.9	98.5	98.6	102.3	100.6	100.9	103.7
96	99.6	93.4	95.1	97.9	98.4	98.6	102.3	100.6	100.9	103.7
97	99.6	93.4	95.1	97.9	98.4	98.6	102.3	100.5	100.9	103.7
98	99.6	93.4	95.1	97.9	98.4	98.6	102.3	100.5	100.8	103.7
99	99.6	93.4	95.0	97.9	98.4	98.6	102.3	100.5	100.9	103.7
100	99.6	93.4	95.0	97.9	98.4	98.6	102.3	100.5	100.9	103.7

Tabel Hasil Ekseprimen Penentuan Jumlah Topik berdasarkan Nilai *Perplexity* – *Stemming Latent Dirichlet Allocation*

Percobaan	Jumlah Topik								
	2	3	4	5	6	7	8	9	10
1	78.9	76.8	74.7	74.6	74.8	72.9	76.4	75.8	76.3
2	78.8	75.5	75.5	74.9	73.0	74.8	74.2	75.0	75.2
3	78.9	76.1	73.7	74.0	73.2	74.6	74.1	74.8	75.1
4	78.9	76.0	74.2	74.1	73.0	74.4	76.0	74.6	77.4
5	78.9	76.8	76.1	73.3	74.6	73.7	74.9	73.9	74.3
6	78.9	76.9	74.3	72.8	73.3	72.1	74.2	75.4	76.4
7	78.9	76.7	74.7	74.0	72.9	74.7	74.5	76.1	75.9
8	80.7	76.5	73.8	72.7	73.9	74.2	75.4	73.9	74.9
9	78.8	76.4	74.5	73.6	73.8	74.1	75.1	75.5	77.0
10	78.8	75.9	73.9	74.0	73.3	74.6	76.5	75.1	76.3
11	78.9	75.7	74.2	73.2	73.2	74.9	75.9	75.3	76.4
12	78.8	76.9	73.8	73.6	74.2	74.3	74.4	74.4	75.2
13	78.8	76.4	73.7	73.1	75.0	74.7	74.3	75.9	77.0
14	78.9	77.0	74.6	74.1	73.2	74.1	74.7	74.7	76.5
15	78.8	77.0	75.6	72.8	73.2	73.6	75.1	75.4	75.2
16	78.8	75.5	73.7	72.2	73.4	73.0	76.1	75.8	76.1
17	78.9	77.5	75.0	76.2	72.6	74.2	75.0	75.3	77.1
18	78.8	76.1	75.2	73.0	73.3	73.9	75.5	76.4	76.1

19	78.9	76.8	74.2	73.9	76.1	73.7	75.6	73.9	75.8
20	78.9	76.8	73.8	72.7	76.4	74.9	75.0	75.1	76.3
21	78.8	76.4	74.2	74.2	73.4	74.9	75.4	76.4	75.1
22	78.9	75.7	74.8	73.2	75.2	75.1	75.9	76.6	75.6
23	78.9	76.5	74.9	72.5	74.1	74.7	76.0	75.3	76.2
24	78.8	77.1	74.2	72.3	73.6	72.7	74.2	76.0	76.3
25	78.8	76.7	74.8	73.7	73.0	75.6	75.5	77.1	76.3
26	79.7	75.8	74.7	73.6	74.8	74.0	76.1	75.1	76.1
27	78.8	75.7	73.8	72.4	74.2	73.6	74.9	76.0	75.4
28	78.9	76.7	73.6	74.9	73.0	74.3	75.8	75.9	74.5
29	78.9	76.7	74.7	75.0	72.9	73.6	75.4	73.9	76.1
30	78.9	77.4	73.7	75.2	75.4	72.8	75.2	74.6	75.2
Rata-Rata	78.947	76.467	74.420	73.660	73.867	74.090	75.243	75.307	75.910
Std Dev	0.36835	0.55232	0.64348	0.96868	1.00184	0.80616	0.70598	0.84484	0.76850

**Tabel Hasil Ekseprimen Penentuan Jumlah Topik berdasarkan Nilai *Perplexity* – Tanpa *Stemming*
*Latent Dirichlet Allocation***

Percobaan	Jumlah Topik								
	2	3	4	5	6	7	8	9	10
1	118.4	116.4	111.4	108.1	109.3	111.0	110.4	113.9	114.3
2	118.3	114.9	108.2	109.9	109.7	111.3	114.0	116.0	113.2
3	118.4	114.2	107.9	107.6	109.7	111.5	111.4	114.9	116.2
4	120.0	114.1	111.2	107.2	107.2	109.1	112.8	115.2	115.6
5	120.9	112.9	109.8	109.0	110.0	109.1	113.6	115.9	115.6
6	118.2	111.4	108.6	109.4	107.2	109.3	112.4	114.3	115.9
7	118.2	114.7	110.0	111.5	110.0	110.9	116.0	116.3	115.3
8	118.3	111.5	109.3	108.0	112.2	113.5	111.2	112.5	115.6
9	118.2	114.4	108.7	112.3	109.9	109.5	113.6	112.7	114.1
10	120.5	112.9	111.4	107.0	109.7	110.3	111.1	115.5	115.0
11	118.3	113.8	109.4	108.1	108.2	116.4	110.4	113.9	114.3
12	118.2	113.0	111.4	110.8	110.1	113.5	115.6	116.1	114.4
13	118.3	114.6	111.3	108.1	109.3	112.2	115.3	113.2	117.7
14	121.3	113.6	108.3	112.7	111.2	110.0	111.9	114.7	116.1
15	118.2	115.0	108.7	108.0	109.8	112.5	114.5	114.5	115.0
16	118.3	112.5	109.1	109.2	110.7	113.3	114.5	115.8	117.7
17	118.2	114.9	108.3	107.3	113.0	110.1	112.5	113.3	115.9
18	118.3	112.1	111.3	108.6	106.1	112.3	114.3	116.0	113.2

19	118.2	111.7	109.2	106.8	108.9	111.9	113.6	114.6	115.3
20	118.2	113.1	110.4	111.3	113.5	112.5	111.7	114.2	113.3
21	118.2	112.7	111.4	106.7	108.1	111.9	115.5	116.7	118.0
22	121.5	112.9	110.5	107.4	113.1	112.0	115.6	118.4	117.0
23	118.4	112.8	111.6	108.4	108.0	110.1	110.4	110.9	113.1
24	118.2	114.5	107.8	106.3	107.5	113.8	112.9	114.3	114.8
25	118.4	113.0	108.4	110.5	108.8	112.2	115.6	112.3	115.8
26	118.2	114.5	108.3	108.1	107.1	114.4	112.6	114.2	116.5
27	118.2	111.6	109.1	109.4	111.3	110.2	114.0	115.8	115.9
28	118.2	114.3	109.3	107.6	111.2	109.6	111.5	117.0	114.9
29	118.2	112.8	110.9	108.9	110.7	110.7	110.0	115.6	115.8
30	118.2	111.7	109.6	108.7	109.7	112.9	117.6	114.7	114.9
Rata-Rata	118.687	113.417	109.693	108.763	109.707	111.600	113.217	114.780	115.347
Std Dev	1.00747	1.26275	1.26598	1.67115	1.84875	1.75539	1.98704	1.57226	1.30033

LAMPIRAN B

LDA Stemming : Daftar Kata Topik 5

Topik 0	Topik 1	Topik 2	Topik 3	Topik 4
0.046*kuliah	0.024*pikir	0.080*pacar	0.041*anak	0.088*teman
0.032*orang_tua	0.022*hidup	0.047*hubung	0.032*rumah	0.073*suka
0.031*sekolah	0.020*laku	0.032*cinta	0.028*suami	0.022*bicara
0.027*ajar	0.019*takut	0.031*sayang	0.026*ayah	0.020*cewek
0.020*masuk	0.018*teman	0.031*putus	0.025*keluarga	0.020*bilang
0.019*kelas	0.014*hati	0.021*mantan	0.024*orang_tua	0.020*cowok
0.016*lulus	0.013*hadap	0.017*laki	0.019*kakak	0.019*sahabat
0.016*pilih	0.012*sakit	0.016*bilang	0.019*uang	0.017*kelas
0.014*teman	0.011*keluarga	0.014*jalan	0.016*adik	0.017*lihat
0.013*guru	0.009*milik	0.014*laku	0.015*tinggal	0.013*kaya
0.012*pikir	0.009*bantu	0.014*wanita	0.014*marah	0.011*kayak
0.012*nilai	0.008*sedih	0.012*temu	0.012*bilang	0.011*laki
0.011*bingung	0.008*sulit	0.012*asa	0.012*istri	0.011*kadang
0.011*jurus	0.008*kadang	0.011*nyata	0.011*pulang	0.010*asa
0.011*tugas	0.008*usaha	0.011*pasang	0.011*menikah	0.010*bikin
0.010*malas	0.008*butuh	0.011*bingung	0.009*makan	0.009*diam
0.009*kampus	0.008*percaya	0.011*hati	0.008*nikah	0.009*main
0.009*hidup	0.008*menang	0.009*lupa	0.007*tidur	0.008*kenal

0.009*izin	0.007*asa	0.009*kenal	0.007*saudara	0.008*ketemu
0.008*kota	0.007*cerita	0.008*cewek	0.007*laku	0.008*marah
0.008*sia	0.007*coba	0.008*kali	0.007*malam	0.008*ajak
0.008*semester	0.007*hilang	0.008*kasih	0.006*suruh	0.008*takut
0.007*cita	0.007*terkadang	0.007*pria	0.006*pergi	0.008*cerita
0.007*nyata	0.007*alami	0.007*sakit	0.006*umur	0.007*bareng
0.007*mati	0.007*bahagia	0.007*hilang	0.006*pakai	0.007*bingung
0.006*juara	0.007*jalan	0.007*ubah	0.006*jam	0.006*langsung
0.006*ambil	0.006*kali	0.007*jarak	0.006*habis	0.006*bawa
0.006*universitas	0.006*bicara	0.007*maaf	0.006*bantu	0.006*malu
0.006*kecewa	0.006*terima_kasih	0.007*terima	0.005*perempuan	0.006*nama
0.006*smp	0.006*lihat	0.007*tinggal	0.005*cari	0.006*biar

Gaussian LDA Stemming : Daftar Kata Topik 5

Topik 0	Topik 1	Topik 2	Topik 3	Topik 4
kuliah	pikir	pacar	anak	teman
orang_tua	hidup	hubung	rumah	suka
sekolah	laku	cinta	suami	bicara
ajar	takut	sayang	ayah	cewek
masuk	teman	putus	keluarga	bilang
kelas	hati	mantan	orang_tua	cowok
lulus	hadap	laki	kakak	sahabat
pilih	sakit	bilang	uang	kelas
teman	keluarga	jalan	adik	lihat
guru	milik	laku	tinggal	kaya
pikir	bantu	wanita	marah	kayak
nilai	sedih	temu	bilang	laki
bingung	sulit	asa	istri	kadang
jurus	kadang	nyata	pulang	asa
tugas	usaha	pasang	menikah	bikin
malas	butuh	bingung	makan	diam
kampus	percaya	hati	nikah	main

hidup	menang	lupa	tidur	kenal
izin	asa	kenal	E	ketemu
kota	cerita	cewek	laku	marah
sia	coba	kali	malam	ajak
semester	hilang	kasih	suruh	takut
cita	terkadang	pria	pergi	cerita
nyata	alami	sakit	umur	bareng
mati	bahagia	hilang	pakai	bingung
juara	jalan	ubah	jam	langsung
ambil	kali	jarak	habis	bawa
universitas	bicara	maaf	bantu	malu
kecewa	terima_kasih	terima	perempuan	nama
smp	lihat	tinggal	cari	biar

LDA Stemming : Daftar Kata Topik 25

Topic 0	Topic 1	Topic 2	Topic 3	Topic 4	Topic 5	Topic 6	Topic 7	Topic 8	Topic 9
hidup	laki	pacar	cinta	bicara	sekolah	keluarga	sosial	suka	marah
sakit	hubung	putus	temu	malu	ajar	orang_tua	sepi	asa	ubah
bahagia	pasang	sayang	wanita	laku	orang_tua	ayah	media	hati	sikap
sedih	jalan	mantan	hubung	nyaman	guru	anak	benci	kadang	diam
hati	perempuan	hubung	asa	jelek	kuliah	tinggal	sibuk	curah	sifat
pikir	jarak	bilang	nyata	anggap	nilai	rumah	peduli	kayak	acuh
menang	serius	move	pria	percaya	pikir	adik	urus	bawa	kaya
laku	jodoh	lupa	laku	olah	sia	laku	dunia	bingung	emosi
lihat	beda	laku	hati	pribadi	cita	beliau	iri	bosan	bicara
tahan	jalin	bingung	milik	gaul	lulus	cerita	tani	pikir	bingung
kuat	komunikasi	jalan	jalan	diam	hidup	nenek	tk	you	hadap
tuhan	laku	selingkuh	kenal	bodoh	kelas	hidup	giat	bilang	bilang
coba	umur	tinggal	sayang	lihat	malas	saudara	anggur	lihat	kasar
harap	restu	ajak	terima	berani	pilih	lahir	abar	malas	maaf
asa	jarang	sesal	bohong	pandang	semangat	tim	terkadang	enak	mengerti
lupa	alas	sakit	takut	lingkungan	kecewa	bilang	bosan	susah	kadang

lelah	bosan	kaya	jujur	ejek	sepakbola	milik	surat	dikit	laku
usaha	tahan	cemburu	kali	fisik	gagal	anggap	alas	beda	egois
sadar	rokok	kali	pikir	introvert	sbmptn	tante	keliling	keras_kepala	sakit_hati
allah	jaga	jujur	kecewa	aneh	raih	dengar	layan	gengsi	menang

Topic 10	Topic 11	Topic 12	Topic 13	Topic 14	Topic 15	Topic 16	Topic 17	Topic 18	Topic 19
paham	suami	teman	pakai	cewek	kuliah	takut	rumah	teman	sahabat
tertawa	anak	cerita	agama	cowok	kampus	pikir	pulang	kelas	biar
hina	menikah	butuh	baca	suka	jurus	tidur	kota	bicara	mimpi
mood	nikah	anggap	tonton	lihat	masuk	hilang	izin	suka	percaya
sulit	istri	dengar	mama	perhatian	tugas	laku	pergi	main	bikin
jaga	umur	ajak	tulis	acuh	ambil	alami	malam	lihat	peka
nomor	selingkuh	sedih	melulu	bilang	semester	stres	jam	sekolah	pura
dewasa	usia	akrab	minum	kenal	mahasiswa	depresi	jalan	diam	cocok
hidup	mertua	milik	hobi	un	dosen	sakit	beli	smp	cemburu
bicara	laku	laku	lihat	loh	bingung	ganggu	tinggal	anak	kangen
just	hamil	beda	buku	malu	ajar	tekan	kantor	ajak	tips
senang	rumah_tangga	susah	bagus	tarik	have	bangun	pindah	bareng	peluk

butuh	tinggal	bahas	film	cari	ketua	bunuh	lolos	bilang	serasa
kadang	ceraai	sikap	cerita	cuek	minat	hadap	suruh	kaya	kode
gila	keluarga	manfaat	tengsin	ketemu	psikologi	negatif	kamar	masuk	ujung
love	pisah	tolak	gambar	ribu	maju	kadang	pagi	kumpul	kayak
mengerti	perempuan	nyaman	laptop	bikin	tingkat	cemas	habis	juara	gampang
mental	abang	pikir	dasar	rada	bakat	berat	bawa	kenal	php
karakter	muda	galau	operasi	wajar	materi	trauma	minggu	langsung	neng
buka	tua	komentar	polisi	minder	olahraga	kali	kos	kelompok	jijik

Topic 20	Topic 21	Topic 22	Topic 23	Topic 24
bilang	bantu	uang	kakak	pilih
ketemu	tolong	butuh	adik	bingung
balas	saran	gara	mati	usaha
maaf	mohon	makan	saudara	cari
pesan	solusi	capek	tanda	lulus
kirin	terima_kasih	bilang	kena	sisi
kenal	kasih	enak	panggil	kuliah
kemarin	kawan	bayar	perempuan	orang_tua

langsung	abai	habis	nama	selesai
minggu	orientasi	kesel	mbak	susah
foto	maaf	suruh	support	hasil
kabar	sempurna	kaya	perilaku	sesuai
tunggu	buang	calon	salat	cepat
nyata	campur	biar	bipolar	terima
telepon_genggam	pulau	bikin	mata	masuk
cerita	matang	bantu	anjing	didik
lihat	coba	pinjam	curhat	alam
situ	nasehat	gaji	belah	kejar
ulang	pasrah	jual	sial	lamar
janji	jenuh	susah	rekan	biaya

Gaussian LDA Stemming : Daftar Kata Topik 25

Topic 0	Topic 1	Topic 2	Topic 3	Topic 4	Topic 5	Topic 6	Topic 7	Topic 8	Topic 9
beres	jati	mengijinkan	lelaki	takdir	gejala	percaya	caci	ips	beres
paksa	konsentras i	kecuali	mantan	rangkin g	problem	takut	punggun g	ptn	paksa
habis	intropeksi	larang	gadis		mengidap	sadar	kunci	kuliah	habis
masak	stabil	merespon	jenis		serba	ragu	buru	smk	masak
kabur	maksimal	memberitahuka n	orang		asmara	dendam	pasar	ipa	kabur
istirahat	ubah	maksud	perempua n		perilaku	setia	berita	swasta	istirahat
bawa	sehat	milik	pacar		emosional	penasara n	kucing	pindah	bawa
jemput	jernih	buah	cowok		pola	kagum	spesial	ip	jemput
mandi	cepat	salam	pria		seks	kasihan	telinga	semester	mandi
sibuk	lari	mbak	cewek		masyaraka t	kalah	puas	sekolah	sibuk
tutup	bangkit	kawan	sahabat		alam	paham	jijik	lulus	tutup
tamu	lepas	kali	teman		fase	kuat	sinis	masuk	tamu

salat	mandiri	kasih	wanita		sudut	lupa	tuduh	juara	salat
berangk t	maju	jarak	laki		contoh	benci	efektif	lolos	berangk t
mobil	pelan	dasar			suku	cemburu	bibi	tim	mobil
pergi	drastis	sekian			finansial	mundur	ribet	universita s	pergi
minum	perlahan	moga			dampak	pisah	ceplos	seleksi	minum
baju	gagal	bos			konflik	ikhlas	bekas	favorit	baju
motor	lambat	urus			anggota	serius	papah	pesantren	motor
jalan	tenaga	sepakat			akibat	curiga	diet		jalan

Topic 10	Topic 11	Topic 12	Topic 13	Topic 14	Topic 15	Topic 16	Topic 17	Topic 18	Topic 19
nenek	risih	sosok	kelak	kasi	bolos	enggan	kado	jaman	seni
tiri	jahat	tulus	syarat	dukung	ekstrakurikuler	gengsi	kode	dulunya	hukum
cerai	gaul	bangga	bukti	bilangin	peringkat	santai	respon	senior	karir
calon	ajar	pendam	karma	menang	fisika	asik	nasehat	osis	sidang
abang	heran	anggap	komitmen	ribut	nasional	minder	tolong	dibully	perawan

kandung	suka	layak	mapan	sumpah	presentasi	konyol	nafkah	beda	wajib
sepupu	nada	normal	sukses	marah	fokus	asing	ampun	buli	sarjana
lahir	sopan	bahagia	doa	ganggu	inggris	gugup	motivasi	siswa	ahli
beliau	over	bebas	rahasia	mesti	olahraga	seru	perhatian	murid	islam
saudara	cuek	luka	kriteria	nasib	lomba	rendah	bantu	enam	konsultasi
ipar	manja	tumbuh	materi	mikirnya	jadwal	ceria	solusi	korban	tamat
keluarga	posesif	kesah	hak	diam	prestasi	interaksi	kabar	orientasi	admin
suami	galak	sederhana	prinsip	omong	rajin	basi	terima	geng	hrd
bayi	laku	hadir	restu	bohong	bagus	murah	maaf	aktif	sbm
adik	ganteng	serasa	informasi	kesel	matematika	kaku	saran	bangku	profesi
tante	pintar	bayang	hamba	tukang	bahasa	drama	hadiah	remaja	alhamdulillah
anak	egois	palsu	tujuan	dikit	organisasi	musik	nasihat	semenjak	mtk
papa	sikap	betapa	proses	sepele	buku	tampil		ketua	teknik
kakak	jelek	kisah	jodoh	tertawa	skripsi	anti		smp	muslim
angkat	bodoh	terkadang	harap	anjing	un	topik		sd	proposal

Topic 20	Topic 21	Topic 22	Topic 23	Topic 24
kost	iseng	kesal	hamil	juni
bandung	ganti	muak	usia	malang
lingkungan	nomor	bimbang	menikah	barusan
yogyakarta	online	capek	umur	sebentar
daerah	kaget	jengkel	nikah	juli
jakarta	pegang	khawatir	muda	januari
zona	kirim	tenang		desember
hotel	status	canggung		selesai
jawa	hapus	lelah		selang
surabaya	lebar	gila		malam
pulau	pasang	frustasi		jam
kampung	isi	lega		esok
posisi	aplikasi	nyaman		puasa
tinggal	nama	bosan		sabtu
kos	video	aduk		senin
toko	baca	gelisah		april
kantor	internet	cemas		maret

situ	instagram	sedih		lebaran
menyukainya	dilupakan	menyukainya	peduli	menyukainya
menghindar	terpikir	menghindar	musuhan	pekerjaan

LAMPIRAN C

Tabel Hasil Pengujian Model pada Aplikasi dengan Metode *Latent Dirichlet Allocation* 5 Topik

Dokumen Ke	Topik	Probabilitas Topik					Kesesuaian
		0	1	2	3	4	
0	2	2.54%	2.52%	2.59%	2.60%	89.75%	Tidak
1	3	4.00%	43.94%	4.12%	23.22%	24.73%	Tidak
2	2	1.67%	26.60%	68.33%	1.70%	1.70%	Ya
3	4	2.23%	35.98%	2.24%	21.93%	37.61%	Ya
4	2	1.69%	1.69%	93.16%	1.70%	1.76%	Ya
5	1	16.60%	73.65%	0.00%	0.00%	8.28%	Ya
6	2	1.55%	1.56%	19.15%	1.56%	76.18%	Tidak
7	2	2.86%	47.40%	44.00%	2.87%	2.87%	Tidak
8	2	1.89%	20.62%	64.49%	11.15%	1.86%	Ya
9	4	1.82%	66.32%	10.22%	1.83%	19.81%	Tidak
10	1	14.94%	64.85%	0.00%	19.57%	0.00%	Ya
11	4	2.23%	12.32%	2.27%	2.26%	80.92%	Ya
12	2	1.12%	17.92%	47.89%	1.13%	31.95%	Ya
13	1	0.00%	61.15%	0.00%	36.96%	85.58%	Ya
14	2	12.06%	0.00%	0.00%	0.00%	85.58%	Tidak
15	2	1.03%	27.19%	69.77%	1.01%	1.00%	Ya
16	3	0.00%	32.76%	0.00%	48.98%	16.64%	Ya

17	2	5.02%	57.52%	28.48%	8.44%	0.00%	Tidak
18	2	1.11%	14.52%	68.65%	1.12%	14.60%	Ya
19	2	1.83%	1.83%	92.68%	1.82%	1.84%	Ya
20	2	1.54%	1.58%	80.63%	1.55%	14.70%	Ya
21	1	0.00%	36.81%	31.07%	24.42%	7.34%	Ya
22	2	4.54%	9.51%	54.99%	10.92%	20.04%	Ya
23	2	0.00%	0.00%	49.84%	19.83%	29.42%	Ya
24	4	2.53%	2.50%	2.67%	25.63%	66.67%	Ya
25	4	1.44%	15.08%	1.47%	20.45%	61.56%	Ya
26	4	2.23%	12.32%	2.27%	2.26%	80.92%	Ya
27	3	0.00%	25.46%	0.00%	71.90%	0.00%	Ya
28	4	19.61%	1.03%	1.03%	33.12%	45.21%	Ya
29	3	0.00%	55.60%	0.00%	42.98%	45.75%	Tidak
30	2	0.00%	46.31%	6.78%	0.00%	45.75%	Tidak
31	2	0.00%	38.60%	26.92%	32.63%	0.00%	Tidak
32	4	37.61%	31.19%	0.00%	16.32%	14.42%	Tidak
33	3	0.00%	24.31%	27.20%	47.57%	0.00%	Ya
34	3	0.00%	18.94%	7.50%	69.51%	3.64%	Ya
35	3	1.47%	1.48%	22.18%	73.35%	1.53%	Ya
36	2	2.50%	2.50%	71.59%	2.52%	20.89%	Ya
37	1	0.00%	92.20%	5.38%	0.00%	0.00%	Ya
38	2	1.34%	1.35%	20.53%	49.79%	27.00%	Tidak
39	3	2.50%	2.55%	17.67%	74.79%	2.50%	Ya

40	2	0.00%	18.34%	44.06%	19.59%	17.04%	Ya
41	4	0.00%	12.84%	56.06%	0.00%	29.48%	Tidak
42	0	49.09%	32.67%	15.34%	1.47%	1.44%	Ya
43	2	1.28%	21.01%	75.14%	1.27%	1.30%	Ya
44	2	27.96%	1.83%	1.85%	19.90%	48.47%	Tidak
45	2	0.00%	0.00%	24.83%	14.22%	59.17%	Tidak
46	4	19.17%	16.04%	0.00%	0.00%	63.71%	Ya
47	4	3.37%	3.50%	3.45%	28.19%	61.50%	Ya
48	0	35.81%	38.54%	0.00%	23.73%	0.00%	Tidak
49	2	0.00%	58.85%	21.72%	0.00%	17.99%	Tidak
50	2	1.83%	1.84%	67.31%	12.87%	16.16%	Ya
51	2	1.44%	1.47%	80.73%	1.45%	14.91%	Ya
52	0	1.47%	47.36%	1.50%	48.23%	1.45%	Tidak
53	2	2.26%	2.27%	90.93%	2.30%	2.25%	Ya
54	2	0.00%	0.00%	80.06%	17.04%	0.00%	Ya
55	0	78.49%	0.00%	0.00%	18.56%	0.00%	Ya
56	3	13.80%	26.30%	4.82%	54.60%	0.00%	Ya
57	2	2.23%	2.22%	46.67%	13.52%	35.36%	Ya
58	2	1.18%	1.19%	68.51%	1.18%	27.94%	Ya
59	1	1.28%	54.79%	1.27%	1.25%	41.41%	Ya
60	4	0.00%	19.49%	12.44%	6.46%	60.92%	Ya
61	0	47.09%	2.04%	2.07%	2.04%	46.76%	Ya
62	2	1.44%	1.45%	74.38%	1.44%	21.30%	Ya

63	2	1.43%	19.11%	42.47%	1.45%	35.55%	Ya
64	3	5.38%	25.02%	0.00%	41.94%	27.46%	Ya
65	2	0.00%	47.91%	50.24%	0.00%	0.00%	Ya
66	2	8.89%	0.00%	34.80%	9.57%	46.36%	Tidak
67	4	17.94%	36.91%	0.00%	0.00%	44.06%	Ya
68	4	0.00%	16.04%	0.00%	17.44%	64.67%	Ya
69	4	1.54%	1.57%	1.54%	1.54%	93.81%	Ya
70	2	9.86%	15.46%	43.69%	30.72%	0.00%	Ya
71	2	2.01%	2.02%	59.14%	2.01%	34.82%	Ya
72	0	42.07%	42.80%	13.73%	0.00%	0.00%	Tidak
73	2	1.43%	19.11%	42.47%	1.45%	35.55%	Ya
74	3	1.00%	13.86%	1.01%	83.13%	1.00%	Ya
75	2	0.00%	47.91%	50.24%	0.00%	0.00%	Ya
76	1	1.06%	55.41%	1.07%	1.06%	41.40%	Ya
77	2	0.00%	6.17%	0.00%	63.07%	29.70%	Tidak
78	1	21.80%	52.94%	0.00%	12.61%	12.18%	Ya
79	2	1.25%	1.26%	41.44%	1.26%	54.79%	Tidak
80	3	0.00%	45.69%	17.51%	31.80%	4.44%	Tidak
81	1	9.70%	55.74%	7.23%	4.77%	22.56%	Ya
82	1	2.92%	88.26%	2.95%	2.93%	2.94%	Ya
83	4	44.23%	1.38%	1.35%	1.34%	51.70%	Ya
84	3	0.00%	15.65%	5.56%	77.96%	0.00%	Ya
85	4	1.35%	1.34%	1.43%	1.35%	94.53%	Ya

86	3	0.00%	20.13%	0.00%	59.73%	19.22%	Ya
87	2	10.17%	0.00%	14.94%	0.00%	73.51%	Tidak
88	2	0.00%	48.00%	49.45%	0.00%	0.00%	Ya
89	0	94.86%	1.32%	1.26%	1.27%	1.29%	Ya
90	2	2.01%	41.55%	52.29%	2.03%	2.12%	Ya
91	0	1.70%	84.41%	1.68%	1.71%	10.51%	Tidak
92	3	0.00%	30.52%	11.80%	57.41%	81.78%	Ya
93	2	0.00%	0.00%	15.58%	0.00%	81.78%	Tidak
94	2	1.18%	1.18%	32.67%	1.20%	63.77%	Tidak
95	2	1.07%	1.07%	95.71%	1.08%	1.07%	Ya
96	2	1.43%	1.49%	1.46%	1.44%	94.18%	Tidak
97	3	1.06%	1.10%	1.07%	89.45%	7.33%	Ya
98	4	0.00%	33.70%	0.00%	17.02%	48.39%	Ya
99	3	2.00%	32.16%	2.07%	61.76%	2.01%	Ya
100	1	1.83%	54.87%	1.84%	39.60%	1.86%	Ya
101	4	1.83%	1.83%	1.85%	1.82%	92.67%	Ya
102	3	2.52%	45.01%	2.58%	47.34%	2.54%	Ya
103	3	0.00%	0.00%	41.45%	56.37%	39.79%	Ya
104	2	0.00%	25.54%	33.89%	0.00%	39.79%	Tidak
105	3	0.00%	32.14%	0.00%	58.83%	8.04%	Ya
106	0	1.15%	27.09%	18.66%	51.98%	1.13%	Tidak
107	3	7.86%	9.01%	24.79%	57.16%	1.18%	Ya
108	2	1.18%	1.21%	48.70%	1.19%	47.72%	Ya

109	0	49.72%	18.18%	0.00%	27.67%	4.18%	Ya
110	3	6.56%	25.43%	8.00%	53.96%	6.06%	Ya
111	0	68.87%	1.17%	6.46%	1.11%	22.39%	Ya
112	2	0.00%	24.90%	36.74%	24.11%	13.51%	Ya
113	3	14.17%	1.11%	1.11%	82.46%	1.14%	Ya
114	1	24.90%	40.47%	0.00%	33.49%	0.00%	Ya
115	3	0.00%	10.04%	17.78%	51.46%	20.07%	Ya
116	0	23.82%	7.72%	66.67%	0.00%	0.00%	Tidak
117	0	46.09%	1.30%	1.26%	50.08%	1.27%	Tidak
118	1	35.64%	2.27%	44.30%	2.30%	15.49%	Tidak
119	2	0.00%	23.17%	55.13%	0.00%	20.52%	Ya
120	4	1.36%	14.08%	1.37%	1.36%	81.83%	Ya
121	2	3.34%	3.36%	86.62%	3.34%	3.34%	Ya
122	0	52.09%	35.48%	1.39%	1.35%	9.69%	Ya
123	0	39.89%	46.55%	1.08%	11.41%	1.07%	Tidak
124	3	9.53%	0.00%	0.00%	52.74%	36.60%	Ya
125	3	7.41%	7.71%	0.00%	76.83%	7.80%	Ya
126	0	46.31%	17.27%	5.68%	20.24%	10.50%	Ya
127	3	1.20%	95.22%	1.19%	1.20%	1.19%	Tidak
128	2	2.54%	2.52%	71.63%	2.53%	20.78%	Ya
129	2	2.27%	2.23%	91.06%	2.22%	2.22%	Ya
130	4	2.23%	2.24%	2.24%	2.24%	91.05%	Ya
131	2	4.04%	8.29%	58.09%	29.00%	0.00%	Ya

132	2	22.43%	11.96%	45.99%	19.33%	0.00%	Ya
133	2	0.00%	8.67%	8.88%	4.45%	77.59%	Tidak
134	1	2.48%	52.88%	20.93%	11.26%	12.46%	Ya
135	3	0.00%	11.75%	8.25%	55.42%	24.30%	Ya
136	0	77.89%	1.09%	18.87%	1.09%	1.07%	Ya
137	4	1.12%	17.81%	1.14%	1.15%	78.77%	Ya
138	2	4.62%	13.34%	80.68%	0.00%	0.00%	Ya
139	1	9.32%	37.24%	14.52%	24.34%	14.59%	Ya
140	0	51.37%	39.93%	2.87%	2.89%	2.94%	Ya
141	0	62.61%	1.23%	21.64%	13.29%	1.23%	Ya
142	2	1.44%	1.45%	74.38%	1.44%	21.30%	Ya
143	0	91.07%	2.23%	2.24%	2.23%	2.23%	Ya
144	1	12.97%	46.63%	0.00%	28.49%	11.29%	Ya
145	2	1.25%	1.26%	17.64%	1.26%	78.59%	Tidak
146	3	7.77%	52.19%	0.00%	38.89%	0.00%	Tidak
147	2	1.01%	8.47%	75.18%	14.32%	1.02%	Ya
148	2	0.00%	3.84%	87.25%	0.00%	7.94%	Ya
149	1	16.66%	28.27%	3.81%	50.96%	0.00%	Tidak
150	2	1.13%	54.06%	42.58%	1.12%	1.12%	Tidak
151	2	25.83%	0.00%	59.62%	13.29%	0.00%	Ya
152	0	23.85%	14.64%	13.55%	38.90%	9.06%	Tidak
153	4	0.00%	40.89%	0.00%	3.97%	54.01%	Ya
154	4	0.00%	7.21%	0.00%	8.99%	81.88%	Ya

155	2	6.67%	37.54%	53.76%	1.03%	1.01%	Ya
156	0	60.66%	14.19%	11.68%	13.15%	0.00%	Ya
157	1	0.00%	64.79%	0.00%	32.89%	0.00%	Ya
158	3	1.00%	44.81%	1.02%	52.16%	1.01%	Ya
159	3	1.43%	47.90%	12.32%	31.75%	6.60%	Tidak
160	1	38.49%	50.10%	0.00%	0.00%	10.05%	Ya
161	2	0.00%	0.00%	69.41%	21.46%	7.29%	Ya
162	0	73.07%	1.28%	1.28%	1.32%	23.06%	Ya
163	3	21.43%	20.70%	0.00%	56.92%	25.91%	Ya
164	3	0.00%	17.74%	58.55%	0.00%	22.10%	Ya
165	2	0.00%	26.12%	48.63%	0.00%	23.98%	Ya
166	2	1.01%	1.01%	25.04%	1.01%	71.92%	Ya
167	2	38.09%	19.13%	1.21%	40.39%	1.19%	Tidak
168	3	0.00%	51.75%	0.00%	34.68%	12.13%	Ya
169	1	63.27%	0.00%	33.79%	50.61%	0.00%	Ya
170	0	0.00%	47.90%	0.00%	50.61%	0.00%	Ya
171	0	0.00%	45.00%	53.27%	0.00%	0.00%	Tidak
172	2	0.00%	14.96%	10.59%	67.57%	6.06%	Ya
173	3	0.00%	9.14%	0.00%	72.30%	17.47%	Ya
174	3	4.97%	7.15%	36.24%	36.68%	14.96%	Ya
175	3	1.44%	78.89%	1.46%	1.45%	16.78%	Ya
176	1	0.00%	26.82%	66.23%	5.84%	0.00%	Ya
177	2	12.90%	10.78%	0.00%	49.33%	26.62%	Ya

178	3	0.00%	0.00%	54.14%	6.50%	38.16%	Ya
179	2	72.53%	0.00%	0.00%	25.57%	0.00%	Ya
180	0	0.00%	62.62%	32.55%	3.76%	0.00%	Ya
181	2	5.26%	6.33%	46.21%	13.30%	28.90%	Tidak
182	2	1.26%	33.04%	9.10%	55.33%	1.27%	Ya
183	3	0.00%	30.94%	0.00%	66.17%	0.00%	Ya
184	3	41.74%	35.66%	6.75%	2.26%	13.58%	Ya
185	0	11.39%	1.57%	31.19%	1.57%	54.28%	Ya
186	2	43.35%	25.92%	0.00%	29.27%	0.00%	Tidak
187	3	3.81%	11.06%	0.00%	34.13%	50.46%	Tidak
188	4	1.19%	1.19%	29.85%	1.19%	66.59%	Ya
189	2	23.58%	45.59%	8.50%	9.80%	12.52%	Tidak
190	1	4.50%	19.36%	65.05%	0.00%	10.81%	Ya
191	2	2.26%	26.11%	2.27%	67.11%	2.24%	Ya
192	3	1.13%	24.40%	1.12%	1.12%	72.23%	Ya
193	4	0.00%	0.00%	16.59%	4.34%	78.16%	Ya
194	2	63.56%	22.17%	0.00%	0.00%	12.97%	Tidak
195	0	91.07%	2.23%	2.24%	2.23%	2.23%	Ya
196	0	22.67%	3.98%	37.05%	0.00%	35.91%	Ya
197	2	1.26%	52.53%	1.26%	43.65%	1.30%	Ya
198	3	1.12%	31.22%	22.73%	1.13%	43.80%	Tidak
199	2	0.00%	15.31%	53.03%	0.00%	30.16%	Tidak
200	2	46.68%	1.82%	1.83%	47.86%	1.82%	Ya

201	2	0.00%	8.21%	0.00%	89.96%	0.00%	Tidak
202	3	21.86%	64.97%	1.44%	1.47%	10.27%	Ya
203	1	1.01%	1.01%	65.35%	1.01%	31.63%	Ya
204	2	1.38%	24.86%	1.36%	71.04%	1.36%	Ya
205	3	9.64%	32.30%	28.65%	20.50%	8.93%	Ya
206	2	0.00%	0.00%	60.20%	5.97%	32.72%	Tidak
207	2	2.01%	14.34%	52.88%	2.02%	28.75%	Ya
208	2	76.67%	10.73%	1.56%	9.49%	1.55%	Ya
209	0	1.86%	1.84%	1.83%	1.83%	92.65%	Ya
210	4	62.80%	0.00%	0.00%	25.92%	10.29%	Ya
211	0	0.00%	43.01%	54.75%	0.00%	0.00%	Ya
212	2	4.90%	0.00%	26.06%	48.48%	19.95%	Ya
213	3	4.00%	43.94%	4.12%	23.22%	24.73%	Ya

Tabel Hasil Pengujian Model pada Aplikasi dengan Metode *Gaussian Latent Dirichlet Allocation* 25 Topik

No	Label	Top Topic					Result 5 Top Topic					Kesesuaian
		1	2	3	4	5	1	2	3	4	5	
1	22	4	13	15	22	12	0	0	0	1	0	Ya
2	8	13	23	5	11	2	0	0	0	0	0	Tidak
3	8	5	8	14	15	13	0	1	0	0	0	Ya
4	15	4	5	23	7	21	0	0	0	0	0	Tidak
5	15	15	5	18	14	4	1	0	0	0	0	Ya
6	13	5	23	12	13	21	0	0	0	1	0	Ya
7	23	5	4	15	23	19	0	0	0	1	0	Ya
8	23	23	13	8	-	-	1	0	0	0	0	Ya
9	8	14	4	9	5	7	0	0	0	0	0	Tidak
10	5	5	8	4	17	13	1	0	0	0	0	Ya
11	15	13	15	5	17	10	0	1	0	0	0	Ya
12	15	15	22	5	13	23	1	0	0	0	0	Ya
13	5	8	5	4	13	18	0	1	0	0	0	Ya
14	4	4	1	8	23	21	1	0	0	0	0	Ya
15	8	5	19	15	12	4	0	0	0	0	0	Tidak

16	8	8	4	23	13	9	1	0	0	0	0	Ya
17	11	12	8	11	13	9	0	0	1	0	0	Ya
18	13	7	4	14	8	9	0	0	0	0	0	Tidak
19	8	23	8	22	17	4	0	1	0	0	0	Ya
20	8	8	5	23	-	-	1	0	0	0	0	Ya
21	8	4	23	13	15	3	0	0	0	0	0	Tidak
22	15	5	12	4	23	15	0	0	0	0	1	Ya
23	8	4	22	15	7	2	0	0	0	0	0	Tidak
24	5	5	14	11	9	19	1	0	0	0	0	Ya
25	5	15	4	5	22	12	0	0	1	0	0	Ya
26	13	15	4	5	9	13	0	0	0	0	1	Ya
27	5	15	22	5	13	23	0	0	1	0	0	Ya
28	5	15	11	8	9	22	0	0	0	0	0	Tidak
29	5	9	12	4	16	3	0	0	0	0	0	Tidak
30	15	23	8	11	1	15	0	0	0	0	1	Ya
31	8	12	13	8	23	4	0	0	1	0	0	Ya
32	13	11	13	15	12	8	0	1	0	0	0	Ya
33	5	16	15	13	3	8	0	0	0	0	0	Tidak
34	11	4	15	11	9	24	0	0	1	0	0	Ya

35	13	11	13	23	21	5	0	1	0	0	0	Ya
36	8	11	5	8	15	23	0	0	1	0	0	Ya
37	8	8	4	5	12	-	1	0	0	0	0	Ya
38	5	13	5	23	8	17	0	1	0	0	0	Ya
39	15	15	2	23	5	21	1	0	0	0	0	Ya
40	13	11	13	5	8	-	0	1	0	0	0	Ya
41	15	15	2	8	4	23	1	0	0	0	0	Ya
42	5	5	4	8	15	13	1	0	0	0	0	Ya
43	13	14	8	20	13	10	0	0	0	1	0	Ya
44	13	5	8	2	13	3	0	0	0	1	0	Ya
45	5	5	23	4	21	8	1	0	0	0	0	Ya
46	8	12	5	8	23	13	0	0	1	0	0	Ya
47	15	15	17	10	4	19	1	0	0	0	0	Ya
48	5	5	21	12	-	-	1	0	0	0	0	Ya
49	13	13	17	23	7	21	1	0	0	0	0	Ya
50	8	8	12	5	14	9	1	0	0	0	0	Ya
51	13	12	18	4	5	8	0	0	0	0	0	Tidak
52	8	5	12	4	8	17	0	0	0	1	0	Ya
53	4	8	1	23	20	7	0	0	0	0	0	Tidak

54	23	13	15	23	21	8	0	0	1	0	0	Ya
55	8	4	8	12	13	20	0	1	0	0	0	Ya
56	10	10	18	4	9	20	1	0	0	0	0	Ya
57	15	4	11	15	14	5	0	0	1	0	0	Ya
58	7	5	21	15	-	-	0	0	0	0	0	Tidak
59	8	8	5	4	12	13	1	0	0	0	0	Ya
60	17	12	5	3	17	4	0	0	0	1	0	Ya
61	15	4	12	15	5	22	0	0	1	0	0	Ya
62	15	2	3	4	23	16	0	0	0	0	0	Tidak
63	8	8	14	4	13	15	1	0	0	0	0	Ya
64	8	5	23	8	13	3	0	0	1	0	0	Ya
65	13	15	11	13	12	8	0	0	1	0	0	Ya
66	13	14	2	13	5	8	0	0	1	0	0	Ya
67	5	15	4	8	5	9	0	0	0	1	0	Ya
68	5	4	5	7	19	21	0	1	0	0	0	Ya
69	15	15	19	4	3	5	1	0	0	0	0	Ya
70	5	5	15	4	8	12	1	0	0	0	0	Ya
71	13	4	5	2	9	1	0	0	0	0	0	Tidak
72	8	15	5	3	23	8	0	0	0	0	1	Ya

73	13	4	20	14	18	9	0	0	0	0	0	Tidak
74	13	5	23	8	13	3	0	0	0	1	0	Ya
75	11	11	4	24	21	14	1	0	0	0	0	Ya
76	8	14	2	13	5	8	0	0	0	0	1	Ya
77	15	5	13	23	12	15	0	0	0	0	1	Ya
78	11	11	15	5	12	4	1	0	0	0	0	Ya
79	13	23	5	9	13	12	0	0	0	1	0	Ya
80	13	15	5	9	13	8	0	0	0	1	0	Ya
81	15	13	8	15	11	24	0	0	1	0	0	Ya
82	4	4	17	9	5	8	1	0	0	0	0	Ya
83	15	8	13	15	23	7	0	0	1	0	0	Ya
84	5	5	15	9	21	17	1	0	0	0	0	Ya
85	11	11	4	5	8	16	1	0	0	0	0	Ya
86	5	15	5	12	9	13	0	1	0	0	0	Ya
87	15	11	15	8	5	24	0	1	0	0	0	Ya
88	15	15	5	22	12	8	1	0	0	0	0	Ya
89	8	13	8	5	15	9	0	1	0	0	0	Ya
90	10	4	10	16	5	7	0	1	0	0	0	Ya
91	8	23	8	13	5	21	0	1	0	0	0	Ya

92	4	19	14	4	23	20	0	0	1	0	0	Ya
93	11	11	8	4	21	23	1	0	0	0	0	Ya
94	15	5	15	4	19	12	0	1	0	0	0	Ya
95	15	15	4	5	9	8	1	0	0	0	0	Ya
96	8	8	4	24	5	12	1	0	0	0	0	Ya
97	8	12	5	13	15	23	0	0	0	0	0	Tidak
98	15	2	21	5	23	12	0	0	0	0	0	Tidak
99	15	9	5	23	4	17	0	0	0	0	0	Tidak
100	11	11	20	2	24	3	1	0	0	0	0	Ya
101	11	9	23	11	13	5	0	0	1	0	0	Ya
102	13	5	22	12	17	13	0	0	0	0	1	Ya
103	23	24	23	3	11	12	0	1	0	0	0	Ya
104	13	11	4	5	13	17	0	0	0	1	0	Ya
105	13	4	8	5	12	2	0	0	0	0	0	Tidak
106	13	11	12	15	13	8	0	0	0	1	0	Ya
107	4	4	8	1	22	10	1	0	0	0	0	Ya
108	11	11	8	4	24	15	1	0	0	0	0	Ya
109	8	8	4	13	23	9	1	0	0	0	0	Ya
110	10	4	11	9	5	8	0	0	0	0	0	Tidak

111	11	11	15	2	9	4	1	0	0	0	0	Ya
112	15	20	12	4	16	15	0	0	0	0	1	Ya
113	8	21	23	9	8	12	0	0	0	1	0	Ya
114	15	2	1	15	11	12	0	0	1	0	0	Ya
115	4	20	17	4	24	14	0	0	1	0	0	Ya
116	23	11	23	4	9	12	0	1	0	0	0	Ya
117	4	4	8	9	22	20	1	0	0	0	0	Ya
118	13	10	22	11	20	9	0	0	0	0	0	Tidak
119	8	5	22	8	20	2	0	0	1	0	0	Ya
120	15	5	23	13	4	15	0	0	0	0	1	Ya
121	8	5	8	12	13	15	0	1	0	0	0	Ya
122	8	5	4	12	-	-	0	0	0	0	0	Tidak
123	10	4	9	7	12	16	0	0	0	0	0	Tidak
124	15	4	16	20	15	13	0	0	0	1	0	Ya
125	11	5	4	11	15	2	0	0	1	0	0	Ya
126	23	11	2	23	21	15	0	0	1	0	0	Ya
127	10	10	8	4	5	23	1	0	0	0	0	Ya
128	23	4	7	2	9	13	0	0	0	0	0	Tidak
129	8	8	22	12	2	-	1	0	0	0	0	Ya

130	8	5	8	13	9	12	0	1	0	0	0	Ya
131	5	15	12	23	5	-	0	0	0	1	0	Ya
132	8	5	4	11	15	8	0	0	0	0	1	Ya
133	8	8	4	5	22	21	1	0	0	0	0	Ya
134	8	22	4	5	15	12	0	0	0	0	0	Tidak
135	24	4	8	13	17	23	0	0	0	0	0	Tidak
136	11	11	15	22	4	9	1	0	0	0	0	Ya
137	8	19	10	16	8	18	0	0	0	1	0	Ya
138	13	5	17	4	13	3	0	0	0	1	0	Ya
139	15	5	22	8	2	12	0	0	0	0	0	Tidak
140	4	5	4	24	12	23	0	1	0	0	0	Ya
141	10	17	4	9	16	-	0	0	0	0	0	Tidak
142	4	4	22	1	14	20	1	0	0	0	0	Ya
143	8	8	14	4	13	15	1	0	0	0	0	Ya
144	20	4	20	14	8	18	0	1	0	0	0	Ya
145	15	11	23	13	4	12	0	0	0	0	0	Tidak
146	5	5	22	15	12	2	1	0	0	0	0	Ya
147	11	9	11	7	2	23	0	1	0	0	0	Ya
148	13	8	4	22	9	13	0	0	0	0	1	Ya

149	15	5	8	4	12	15	0	0	0	0	1	Ya
150	4	4	9	14	20	1	1	0	0	0	0	Ya
151	23	23	8	7	13	3	1	0	0	0	0	Ya
152	8	5	8	10	11	4	0	1	0	0	0	Ya
153	4	4	20	22	9	12	1	0	0	0	0	Ya
154	5	4	19	5	13	7	0	0	1	0	0	Ya
155	5	15	5	14	12	2	0	1	0	0	0	Ya
156	8	3	23	4	15	8	0	0	0	0	1	Ya
157	20	4	10	8	15	20	0	0	0	0	1	Ya
158	11	4	23	8	11	21	0	0	0	1	0	Ya
159	23	11	13	23	18	5	0	0	1	0	0	Ya
160	15	15	18	9	13	23	1	0	0	0	0	Ya
161	8	4	13	20	10	8	0	0	0	0	1	Ya
162	8	8	24	5	2	9	1	0	0	0	0	Ya
163	10	10	8	17	3	4	1	0	0	0	0	Ya
164	4	4	11	21	5	1	1	0	0	0	0	Ya
165	11	15	17	4	24	11	0	0	0	0	1	Ya
166	8	15	12	4	23	8	0	0	0	0	1	Ya
167	8	8	14	12	15	5	1	0	0	0	0	Ya

168	23	15	22	4	8	9	0	0	0	0	0	Tidak
169	4	18	21	4	5	23	0	0	1	0	0	Ya
170	11	15	9	18	4	5	0	0	0	0	0	Tidak
171	15	10	18	5	8	11	0	0	0	0	0	Tidak
172	4	4	1	16	14	15	1	0	0	0	0	Ya
173	15	13	4	23	5	22	0	0	0	0	0	Tidak
174	11	11	22	9	13	5	1	0	0	0	0	Ya
175	8	9	8	11	15	4	0	1	0	0	0	Ya
176	11	11	4	24	23	5	1	0	0	0	0	Ya
177	5	17	5	8	12	4	0	1	0	0	0	Ya
178	8	8	5	23	24	2	1	0	0	0	0	Ya
179	15	15	4	9	10	5	1	0	0	0	0	Ya
180	8	5	4	15	2	22	0	0	0	0	0	Tidak
181	20	10	9	20	11	4	0	0	1	0	0	Ya
182	8	4	13	8	5	7	0	0	1	0	0	Ya
183	8	4	5	9	8	22	0	0	0	1	0	Ya
184	15	11	15	2	23	5	0	1	0	0	0	Ya
185	11	11	4	15	21	12	1	0	0	0	0	Ya
186	20	4	10	8	5	13	0	0	0	0	0	Tidak

187	15	19	17	12	9	5	0	0	0	0	0	Tidak
188	8	4	11	16	8	9	0	0	0	1	0	Ya
189	23	15	13	5	2	23	0	0	0	0	1	Ya
190	8	5	12	8	4	-	0	0	1	0	0	Ya
191	10	18	8	10	4	12	0	0	1	0	0	Ya
192	8	5	4	8	22	23	0	0	1	0	0	Ya
193	11	11	4	21	8	19	1	0	0	0	0	Ya
194	5	5	15	2	13	17	1	0	0	0	0	Ya
195	8	12	5	8	4	17	0	0	1	0	0	Ya
196	16	16	3	4	10	5	1	0	0	0	0	Ya
197	20	4	20	14	8	18	0	1	0	0	0	Ya
198	8	5	4	8	19	15	0	0	1	0	0	Ya
199	11	11	13	5	4	15	1	0	0	0	0	Ya
200	8	12	17	23	13	15	0	0	0	0	0	Tidak
201	8	5	4	15	17	8	0	0	0	0	1	Ya
202	13	1	14	20	24	10	0	0	0	0	0	Tidak
203	13	1	11	5	14	13	0	0	0	0	1	Ya
204	15	9	3	20	13	15	0	0	0	0	1	Ya
205	8	5	8	15	4	-	0	1	0	0	0	Ya

206	11	11	15	24	3	23	1	0	0	0	0	Ya
207	13	4	5	9	8	3	0	0	0	0	0	Tidak
208	8	5	4	17	8	15	0	0	0	1	0	Ya
209	5	4	8	12	5	2	0	0	0	1	0	Ya
210	20	4	8	10	5	11	0	0	0	0	0	Tidak
211	5	5	19	4	22	12	1	0	0	0	0	Ya
212	20	10	19	20	16	4	0	0	1	0	0	Ya
213	8	13	23	5	4	8	0	0	0	0	1	Ya
214	12	5	11	4	12	24	0	0	0	1	0	Ya

Tabel Hasil Pengujian Model pada Aplikasi dengan Metode *Latent Dirichlet Allocation* 25 Topik

No	Label	Top Topic					Result 5 Top Topic					Kesesuaian
		1	2	3	4	5	1	2	3	4	5	
1	9	20	24	-	-	-	0	0	0	0	0	Tidak
2	1	16	23	8	9	-	0	0	0	0	0	Tidak
3	3	21	8	20	15	-	0	0	0	0	0	Tidak
4	9	23	21	4	5	13	0	0	0	0	0	Tidak
5	3	12	11	15	13	-	0	0	0	0	0	Tidak
6	21	22	18	16	1	3	0	0	0	0	0	Tidak
7	3	16	19	11	13	14	0	0	0	0	0	Tidak
8	9	8	-	-	-	-	0	0	0	0	0	Tidak
9	2	21	9	24	15	22	0	0	0	0	0	Tidak
10	3	18	17	21	22	8	0	0	0	0	0	Tidak
11	9	22	5	20	10	17	0	0	0	0	0	Tidak
12	11	20	3	12	16	-	0	0	0	0	0	Tidak
13	9	8	12	9	16	6	0	0	1	0	0	Ya
14	17	15	5	18	23	8	0	0	0	0	0	Tidak
15	1	16	19	25	4	2	0	0	0	0	0	Tidak
16	11	21	8	11	14	-	0	0	1	0	0	Ya

17	9	1	24	20	22	18	0	0	0	0	0	Tidak
18	17	22	8	21	18	15	0	0	0	0	0	Ya
19	1	6	21	19	11	18	0	0	0	0	0	Tidak
20	1	6	12	11	19	-	0	0	0	0	0	Tidak
21	1	17	8	21	19	-	0	0	0	0	0	Tidak
22	3	9	22	18	6	8	0	0	0	0	0	Tidak
23	3	21	12	5	15	24	0	0	0	0	0	Tidak
24	3	6	9	11	16	19	0	0	0	0	0	Tidak
25	9	16	12	5	24	13	0	0	0	0	0	Tidak
26	17	20	16	10	3	-	0	0	0	0	0	Tidak
27	11	20	3	12	16	-	0	0	0	0	0	Tidak
28	17	5	9	24	7	20	0	0	0	0	0	Tidak
29	9	3	14	20	16	17	0	0	0	0	0	Tidak
30	17	1	18	22	5	20	0	0	0	0	0	Tidak
31	1	19	18	24	9	2	0	0	0	0	0	Tidak
32	9	9	8	21	18	20	1	0	0	0	0	Ya
33	9	3	20	5	18	16	0	0	0	0	0	Tidak
34	17	9	1	6	20	15	0	0	0	0	0	Ya
35	9	1	15	12	11	9	0	0	0	0	1	Ya

36	11	1	18	7	25	15	0	0	0	0	0	Tidak
37	3	6	11	8	19	20	0	0	0	0	0	Tidak
38	1	23	22	17	11	2	0	0	0	0	0	Tidak
39	3	5	20	11	13	-	0	0	0	0	0	Tidak
40	1	1	17	-	-	-	1	0	0	0	0	Ya
41	17	12	21	11	18	22	0	0	0	0	0	Tidak
42	3	16	8	21	25	11	0	0	0	0	0	Tidak
43	9	22	15	2	16	10	0	0	0	0	0	Tidak
44	2	11	21	15	7	4	0	0	0	0	0	Tidak
45	1	16	19	7	23	5	0	0	0	0	0	Tidak
46	3	19	8	15	25	1	0	0	0	0	0	Tidak
47	1	16	18	2	15	10	0	0	0	0	0	Tidak
48	3	16	1	-	-	-	0	0	0	0	0	Tidak
49	11	3	18	5	10	15	0	0	0	0	0	Tidak
50	3	21	15	20	18	25	0	0	0	0	0	Tidak
51	11	11	9	15	12	-	1	0	0	0	0	Ya
52	1	6	11	17	8	19	0	0	0	0	0	Tidak
53	9	5	22	21	23	-	0	0	0	0	0	Tidak
54	1	8	20	21	17	-	0	0	0	0	0	Tidak

55	3	8	21	9	22	-	0	0	0	0	0	Tidak
56	16	2	15	3	10	7	0	0	0	0	0	Tidak
57	7	1	22	5	2	9	0	0	0	0	0	Tidak
58	13	11	25	5	13	-	0	0	0	1	0	Ya
59	3	21	11	25	13	8	0	0	0	0	0	Tidak
60	13	23	19	22	-	-	0	0	0	0	0	Tidak
61	1	19	20	24	3	5	0	0	0	0	0	Tidak
62	6	15	2	11	12	4	0	0	0	0	0	Tidak
63	3	6	21	8	24	-	0	0	0	0	0	Tidak
64	9	25	20	8	18	6	0	0	0	0	0	Tidak
65	7	1	13	8	22	9	0	0	0	0	0	Tidak
66	3	8	11	7	2	21	0	0	0	0	0	Tidak
67	3	21	13	20	25	11	0	0	0	0	0	Ya
68	11	16	23	22	3	18	0	0	0	0	0	Ya
69	17	13	16	18	10	23	0	0	0	0	0	Ya
70	20	16	17	19	3	-	0	0	0	0	0	Tidak
71	3	21	5	9	24	18	0	0	0	0	0	Tidak
72	3	6	21	25	24	-	0	0	0	0	0	Tidak
73	9	2	7	15	3	6	0	0	0	0	0	Tidak

74	3	25	20	8	18	6	0	0	0	0	0	Tidak
75	17	1	9	15	5	3	0	0	0	0	0	Tidak
76	3	8	11	7	2	21	0	0	0	0	0	Tidak
77	1	16	8	3	-	-	0	0	0	0	0	Tidak
78	7	1	4	20	13	19	0	0	0	0	0	Tidak
79	16	18	3	15	19	17	0	0	0	0	0	Ya
80	3	13	25	21	11	17	0	0	0	0	0	Tidak
81	7	22	9	11	5	3	0	0	0	0	0	Ya
82	21	18	23	20	22	19	0	0	0	0	0	Ya
83	13	15	22	-	-	-	0	0	0	0	0	Tidak
84	6	16	10	2	-	-	0	0	0	0	0	Tidak
85	7	9	1	15	4	20	0	0	0	0	0	Tidak
86	3	16	13	19	12	20	0	0	0	0	0	Tidak
87	7	9	20	23	18	15	0	0	0	0	0	Tidak
88	1	13	12	25	18	22	0	0	0	0	0	Tidak
89	1	8	20	6	16	15	0	0	0	0	0	Tidak
90	16	2	15	7	3	16	0	0	0	0	1	Ya
91	1	8	16	15	23	18	0	0	0	0	0	Tidak
92	9	22	23	3	18	-	0	0	0	0	0	Tidak

93	7	1	8	9	10	18	0	0	0	0	0	Ya
94	3	25	12	21	20	19	0	0	0	0	0	Tidak
95	3	13	20	11	24	21	0	0	0	0	0	Tidak
96	17	11	6	3	21	9	0	0	0	0	0	Tidak
97	1	19	12	23	9	17	0	0	0	0	0	Tidak
98	7	5	22	17	8	21	0	0	0	0	0	Tidak
99	3	17	20	16	13	15	0	0	0	0	0	Tidak
100	7	3	9	15	-	-	0	0	0	0	0	Tidak
101	21	22	1	19	20	-	0	0	0	0	0	Tidak
102	9	12	19	13	16	-	0	0	0	0	0	Tidak
103	1	9	18	-	-	-	0	0	0	0	0	Tidak
104	7	9	21	17	20	14	0	0	0	0	0	Tidak
105	3	21	7	19	6	3	0	0	0	0	1	Ya
106	7	9	1	23	8	19	0	0	0	0	0	Tidak
107	17	15	18	20	9	22	0	0	0	0	0	Tidak
108	7	9	10	6	15	18	0	0	0	0	0	Tidak
109	3	8	17	11	24	13	0	0	0	0	0	Tidak
110	16	2	9	3	10	5	0	0	0	0	0	Tidak
111	7	1	5	10	13	23	0	0	0	0	0	Tidak

112	16	2	17	19	15	21	0	0	0	0	0	Tidak
113	3	21	20	18	13	11	0	0	0	0	0	Tidak
114	9	12	1	5	2	13	0	0	0	0	0	Ya
115	9	15	21	9	2	4	0	0	1	0	0	Ya
116	7	9	18	21	24	19	0	0	0	0	0	Tidak
117	17	12	15	24	2	20	0	0	0	0	0	Ya
118	16	10	22	24	1	3	0	0	0	0	0	Tidak
119	17	2	21	13	24	4	0	0	0	0	0	Tidak
120	3	23	8	20	21	11	0	0	0	0	0	Tidak
121	3	19	12	23	10	4	0	0	0	0	0	Tidak
122	3	6	11	-	-	-	0	0	0	0	0	Tidak
123	6	2	12	17	22	23	0	0	0	0	0	Tidak
124	6	24	2	14	22	5	0	0	0	0	0	Tidak
125	9	13	16	5	9	1	0	0	0	1	0	Ya
126	7	5	9	1	13	20	0	0	0	0	0	Tidak
127	16	3	2	5	15	12	0	0	0	0	0	Ya
128	7	8	22	1	24	10	0	0	0	0	0	Tidak
129	1	19	6	-	-	-	0	0	0	0	0	Tidak
130	1	8	24	11	-	-	0	0	0	0	0	Tidak

131	3	25	19	16	20	-	0	0	0	0	0	Tidak
132	3	11	9	20	22	8	0	0	0	0	0	Tidak
133	3	8	21	24	10	5	0	0	0	0	0	Tidak
134	3	24	23	19	20	16	0	0	0	0	0	Tidak
135	7	18	22	9	21	20	0	0	0	0	0	Tidak
136	7	9	24	20	1	8	0	0	0	0	0	Tidak
137	16	2	6	3	15	-	0	0	0	0	0	Tidak
138	13	16	20	1	7	-	0	0	0	0	0	Tidak
139	13	11	23	21	13	9	0	0	0	1	0	Ya
140	13	8	9	10	23	22	0	0	0	0	0	Tidak
141	16	2	15	17	-	-	0	0	0	0	0	Tidak
142	6	14	20	15	24	12	0	0	0	0	0	Tidak
143	3	6	21	8	24	-	0	0	0	0	0	Tidak
144	16	2	15	7	-	-	0	0	0	0	0	Tidak
145	7	1	18	12	17	14	0	0	0	0	0	Tidak
146	9	16	24	6	19	20	0	0	0	0	0	Tidak
147	7	5	18	1	8	15	0	0	0	0	0	Tidak
148	3	21	24	14	6	15	0	0	0	0	0	Tidak
149	3	11	21	6	19	8	0	0	0	0	0	Tidak

150	17	15	5	23	10	1	0	0	0	0	0	Ya
151	3	8	18	21	22	11	0	0	0	0	0	Tidak
152	3	11	10	9	21	-	0	0	0	0	0	Tidak
153	9	5	15	12	17	24	0	0	0	0	0	Tidak
154	9	23	16	20	22	24	0	0	0	0	0	Tidak
155	3	20	23	16	13	5	0	0	0	0	0	Tidak
156	3	8	20	21	11	10	0	0	0	0	0	Tidak
157	16	3	2	1	21	10	0	0	0	0	0	Ya
158	7	1	20	4	9	15	0	0	0	0	0	Tidak
159	7	1	20	18	22	14	0	0	0	0	0	Tidak
160	13	14	19	8	20	5	0	0	0	0	0	Ya
161	16	23	7	17	10	3	0	0	0	0	0	Tidak
162	17	11	9	5	21	25	0	0	0	0	0	Tidak
163	16	2	7	20	1	15	0	0	0	0	0	Tidak
164	9	5	9	15	22	10	0	1	0	0	0	Ya
165	7	9	13	20	15	-	0	0	0	0	0	Tidak
166	3	21	20	19	8	22	0	0	0	0	0	Tidak
167	3	8	20	23	22	24	0	0	0	0	0	Tidak
168	3	20	24	11	13	19	0	0	0	0	0	Tidak

169	17	15	5	9	8	7	0	0	0	0	0	Tidak
170	7	4	20	22	15	1	0	0	0	0	0	Tidak
171	16	2	24	3	7	9	0	0	0	0	0	Tidak
172	17	15	5	9	18	22	0	0	0	0	0	Ya
173	3	8	10	23	11	20	0	0	0	0	0	Tidak
174	7	9	24	22	7	19	0	0	0	1	0	Ya
175	7	24	20	1	16	4	0	0	0	0	0	Tidak
176	7	9	11	13	6	21	0	0	0	0	0	Tidak
177	13	8	16	19	1	-	0	0	0	0	0	Tidak
178	7	8	9	21	11	16	0	0	0	0	0	Ya
179	16	5	20	22	10	16	0	0	0	0	1	Ya
180	3	21	25	16	7	11	0	0	0	0	0	Tidak
181	16	2	1	7	10	23	0	0	0	0	0	Tidak
182	3	8	22	24	15	20	0	0	0	0	0	Tidak
183	3	8	21	24	11	10	0	0	0	0	0	Tidak
184	2	9	22	20	13	-	0	0	0	0	0	Tidak
185	7	9	1	22	3	7	0	0	0	0	1	Ya
186	16	2	16	3	8	23	0	1	0	0	0	Ya
187	2	19	21	16	7	11	0	0	0	0	0	Tidak

188	7	23	1	10	2	22	0	0	0	0	0	Ya
189	17	5	20	16	12	8	0	0	0	0	0	Tidak
190	1	19	12	16	11	8	0	0	0	0	0	Tidak
191	16	15	3	21	20	10	0	0	0	0	0	Tidak
192	3	21	8	12	9	18	0	0	0	0	0	Ya
193	7	1	9	4	-	-	0	0	0	0	0	Tidak
194	11	16	21	10	17	15	0	0	0	0	0	Tidak
195	3	19	8	24	12	16	0	0	0	0	0	Tidak
196	16	3	7	12	9	22	0	0	0	0	0	Ya
197	16	2	15	7	-	-	0	0	0	0	0	Tidak
198	3	12	16	11	19	17	0	0	0	0	0	Ya
199	2	9	22	16	1	17	0	0	0	0	0	Tidak
200	2	19	18	11	20	21	0	0	0	0	0	Tidak
201	3	25	6	21	20	24	0	0	0	0	0	Tidak
202	3	2	15	9	14	-	0	0	0	0	0	Tidak
203	7	1	5	22	14	9	0	0	0	0	0	Tidak
204	16	18	2	10	20	3	0	0	0	0	0	Tidak
205	3	6	16	11	13	-	0	0	0	0	0	Tidak
206	7	9	22	20	4	-	0	0	0	0	0	Tidak

207	3	9	18	5	12	8	0	0	0	0	0	Tidak
208	3	11	6	13	19	21	0	0	0	0	0	Tidak
209	2	21	11	19	-	-	0	0	0	0	0	Tidak
210	16	2	10	9	-	-	0	0	0	0	0	Tidak
211	13	16	24	-	-	-	0	0	0	0	0	Tidak
212	16	3	16	10	20	2	0	1	0	0	0	Ya
213	1	8	12	22	20	18	0	0	0	0	0	Tidak
214	7	9	21	16	24	25	0	0	0	0	0	Tidak